A New Class of Stochastic EM Algorithms. Escaping Local Maxima and Handling Intractable Sampling

Stéphanie Allassonnière, Juliette Chevallier

A New Class of Stochastic EM Algorithms. Escaping Local Maxima and Handling Intractable Sampling[☆]

Stéphanie Allassonnière[b], Juliette Chevallier[a,*]

[a] *Centre de Mathématiques Appliquées, Écoles polytechniqye, Palaiseau, France*
[b] *Centre de Recherche des Cordeliers, Université Paris-Descartes, Paris, France*

Abstract

The expectation-maximization (EM) algorithm is a powerful computational technique for maximum likelihood estimation in incomplete data models. When the expectation step cannot be performed in closed form, a stochastic approximation of EM (SAEM) can be used. The convergence of the SAEM toward critical points of the observed likelihood has been proved and its numerical efficiency has been demonstrated. However, sampling from the posterior distribution may be intractable or have a high computational cost. Moreover, despite appealing features, the limit position of this algorithm can strongly depend on its starting one. To cope with this two issues, we propose here a new stochastic approximation version of the EM in which we do not sample from the exact distribution in the expectation phase of the procedure. We first prove the convergence of this algorithm toward critical points of the observed likelihood. Then, we propose an instantiation of this general procedure to favor convergence toward global maxima. Experiments on synthetic and real data highlight the performance of this algorithm in comparison to the SAEM and the EM when feasible.

Keywords: EM-like algorithm, stochastic approximation, stochastic optimization, tempered distribution, theoretical convergence.

1. Introduction

Although the expectation-maximization (EM) algorithm (Dempster et al., 1977; Wu, 1983) is a very popular and often efficient approach to *maximum* likelihood (or *maximum a posteriori*) estimation in incomplete data models, as it a simple-to-use algorithm, it has one major issue: the computation of the expectation with respect to the conditional distribution. Indeed, in certain situations, the EM algorithm is not applicable because the expectation step cannot be performed in closed form. To overcome this restriction, many different options have been proposed. In this paper, we concentrate our point on stochastic versions of his basic algorithm. The first idea is to replace the expectation step by a sampling of the unobserved data step. We refer to this EM version as the stochastic EM (SEM) algorithm (Celeux and Diebolt, 1985). In particular, in the SEM, only one sample of the latent variable is drawn. A possible generalization of the SEM is the Monte-Carlo EM (MCEM) (Wei and Tanner, 1990; Fort and Moulines, 2003), in which a Monte-Carlo implementation of the expectation in the expectation-step is carried out. Although the MCEM algorithm provides an elegant solution, it requires a constant increase in the amount of simulated data throughout the algorithm and often a huge amount of simulations (Booth and Hobert, 1999; Jank, 2005).

[☆]Ce travail bénéficie d'un financement public Investissement d'avenir, réference ANR-11-LABX-0056-LMH. This work was supported by a public grant as part of the Investissement d'avenir, project reference ANR-11-LABX-0056-LMH.

[*]Corresponding author
Email address: juliette.chevallier@polytechnique.edu (Juliette Chevallier)

In an alternative way, Delyon et al. (1999) proposed to replace the expectation step of the EM algorithm by one iteration of a stochastic approximation procedure, referred to as SAEM, standing for stochastic approximation EM. This variant relies on the stochastic approximation technique of Robbins and Monro (1951). Essentially, the SAEM algorithm obtains an increasingly accurate approximation of the expectation by computing a weighted average of the empirical mean and all the approximations from previous iterations. Hence, all the simulated data contribute to approximate the expectation. A decreasing sequence of step sizes allows to benefit more and more from these preceding iterations while placing less and less emphasis on the last, more imprecise Monte-Carlo approximation. This method is proved to converge toward stationary points of the log-likelihood, without increasing the sample size at each iteration (Delyon et al., 1999). Moreover, under certain conditions on the likelihood, the convergence toward local *maxima* is guaranteed (Delyon et al., 1999). One hope is that in addition to avoiding saddle points and circumventing the computation of the expectation, introducing randomness may enable to escape local *maxima*. However, this is not yet theoretically proved nor numerically illustrated in the literature.

The EM algorithm has a long and rich history, with a vast literature (Meng and Van Dyk, 1997; McLachlan and Krishnan, 2007), and it is still under development (Balakrishnan et al., 2017). We focus here on the stochastic variants of the EM and more precisely on its stochastic approximation. The SAEM is still attracting significant research interest, for example concerning the choice of the step size and the appropriate stopping rule (Jank, 2006). Moreover, its numerical efficiency has been demonstrated in several situations such as in inference in hidden Markov models (Cappé et al., 2005).

All theoretical results regarding the convergence of the SAEM algorithm assume that we can sample from the posterior distribution. But, in practice, it may be intractable or have a high computational cost. When sampling from the posterior distribution is prohibitive, one may want to shift to variational inference (Jordan et al., 1999; Wainwright and Jordan, 2008; Blei et al., 2017). However, the variational inference method comes without warranty. Hence, if one requires precise samples, it is preferable to resort to a Markov chain Monte Carlo (MCMC) method. The SAEM algorithm was successfully combined with a MCMC procedure (Kuhn and Lavielle, 2004; Allassonnière et al., 2010), which allowed for a whole field of active research. The MCMC-SAEM algorithm is particularly well suited to the study of non-linear mixed-effect models (Kuhn and Lavielle, 2005) and is constantly improving. For example, improvements have been proposed to take into account missing data (Jiang et al., 2018) or to speed its convergence (Karimi et al., 2020). When the posterior distribution is not analytically available, one can sample from an approximation of the posterior distribution, *e.g.* using approximate Bayesian computational (ABC) methods, also known as likelihood-free techniques (see (Marin et al., 2012) for a review). In this sense and to overcome the sampling issue, Picchini and Samson (2018) have proposed to couple the SAEM algorithm to an ABC step, leading to the ABC-SAEM algorithm. Simulations show that this algorithm can be calibrated to return accurate inference, and in some situations, it can outperform a version of the SAEM incorporating the bootstrap filter. However, Picchini and Samson (2018) do not provide any theoretical guarantee of its convergence.

Moreover, despite appealing features, the limit position of the SAEM algorithm can strongly depend on its initialization. To avoid convergence toward local *maxima*, Lavielle and Moulines (1997) have proposed a simulated annealing version of the SAEM. The main idea was to allow the procedure to better explore the state-space by considering a tempered version of the model. More precisely, assuming that the data are corrupted by an additive Gaussian noise with variance σ^2, at each iteration k of the SAEM algorithm, they consider the "false" model in which the noise variance is equal to $((1 + T_k)\sigma)^2$, where $(T_k)_{k\in\mathbb{N}}$ is a positive sequence of temperatures that decreases slowly toward 0. Therefore, the bigger T_k is, the more the likelihood of the model is flattened and the optimizing sequence can escape easily from local *maxima*, and a fortiori, from saddle points. The simulations gave good results but there were no theoretical guarantee for this procedure. Based on the same idea, Lavielle (2014) has proposed to use the simulated-annealing process as a "trick" to better initialize the SAEM algorithm. This initialization scheme is implemented in the MONOLIX software and gives impressive results on real data (Samson et al., 2006; Lavielle and Mentré, 2007; Chan et al., 2011).

In this paper, we will tackle both issues at once. We propose here a new stochastic approximation version of the EM algorithm where we do not sample from the exact distribution but rather from a distribution that converges to the conditional one along the algorithm iterations. This new procedure allows us to derive

a wide class of SAEM-like algorithms, including the “trick" initialized SAEM of Lavielle (2014) and the ABC-SAEM algorithms (Picchini and Samson, 2018), to cope with intractable or difficult sampling. We refer to this new algorithm as the approximated-SAEM.

This general framework allows us to build a procedure, in the spirit of the simulated annealing version of the SAEM (Lavielle and Moulines, 1997), to favor convergence toward global *maxima*. The simulated annealing algorithm (Kirkpatrick et al., 1983) has already been successful on massive datasets, such as for component separation in astrophysical images (Kuruoğlu et al., 2003). However, neither the pure simulated annealing nor the EM algorithm could fully recover the distribution. Our intuition is that coupling the EM algorithm with simulated annealing optimization strategy can get rid of the limitations that Kuruoğlu et al. (2003) raised.

In that goal, we introduce a sequence of temperatures and we sample from a tempered version of the conditional distribution. Therefore, the conditional likelihood of the model is “flattened” and the optimizing sequence can escape more easily from local *maxima*. We refer to this particular instantiation as the tempering-SAEM. Note that our tempering-SAEM differs from the ones of Lavielle and Moulines (1997) as we do not modify the model but only the sampling-step in the estimation algorithm.

The paper is organized as follows: Section 2 briefly recalls the SAEM algorithm and provides details on the theoretical guarantees surrounding its convergence. In Section 3, we introduce our new stochastic approximation version of the EM algorithm, namely the approximated-SAEM, and prove the convergence of this algorithm toward critical points under similar assumptions as the initial SAEM. The proof of the convergence, by its similarity with the proof of the convergence of the SAEM, highlights the unconstraining nature of the different assumptions and therefore the great applicability of our algorithm. In particular, under the same assumptions about the model's likelihood as those introduced by Delyon et al. (1999), it converges towards local *maxima*. We then introduce the tempering-SAEM and provide a theoretical study of its convergence toward stationary points. We also give an heuristic to its convergence toward “less local" *maxima*. Section 4 is dedicated to experiments. The first application we take into account is the *maximum* likelihood estimation of the parameters of a multivariate Gaussian mixture models. This example supports the previous heuristic discussion and gives intuitions into the behavior of the tempering-SAEM algorithm. The second application is an independent factor analysis. In both applications, we focus on the contribution of the tempering-SAEM in comparison to the SAEM, as the possibility to use approximated sampling is covered by Picchini and Samson (2018).

2. Maximum Likelihood Estimation through EM-Like Algorithms

We use in the sequel the classical terminology of the missing data problem, even though the approaches developed here apply to a more general context.

Let $\mathcal{Y} \subset \mathbb{R}^{n_y}$ denote the set of observations, $\mathcal{Z} \subset \mathbb{R}^{n_z}$ the set of latent variables, and $\Theta \subset \mathbb{R}^{n_\theta}$ the set of admissible parameters. Let μ be a σ-finite positive Borel measure on $\mathcal{Z}$. For sake of simplicity, we will use the notation q for different likelihoods, specifying their variables in brackets. In particular, for all $(y;\theta) \in \mathcal{Y} \times \Theta$, $q(y,\,\cdot\,;\theta)$ is the complete likelihood given the observation y and parameter θ and we assume it to be integrable with respect to the measure μ. As for, we note $q(y;\theta) = \int_{\mathcal{Z}} q(y,z;\theta)\,\mathrm{d}\mu(z)$ the observed likelihood and $q(z|y;\theta) = \frac{q(y,z;\theta)}{q(y;\theta)}$ the posterior distribution of the missing data z given the observed data y. Our goal is to estimate the parameters that maximize the likelihood of the observations of n independent samples of a random variable Y, *i.e.* that maximize the observed data likelihood:

$$\text{Given } \; y_1^n = (y_1,\dots,y_n) \in \mathcal{Y}^n, \qquad \widehat{\theta}_n^{MLE} \in \operatorname*{argmax}_{\theta\in\Theta} q\big(y_1^n;\theta\big)\,.$$

For the sake of readability, y_1^n is noted as y from now on.

Unfortunately, *maximum* likelihood estimation (MLE) problem has generally no closed-form solution. The expectation-maximization (EM) algorithm and its variants are powerful algorithms which have demonstrated their efficiency in practice. Given a current parameter estimate θ_k, the EM algorithm seeks to find the MLE by iteratively applying these two steps:

Expectation: Compute the conditional expected log-likelihood

$$Q(\theta|\theta_k) \;=\; \int_{\mathcal{Z}} \log q(y,z;\theta) q(z|y;\theta_k)\, \mathrm{d}z \;=\; \mathbb{E}\left[\log q(Z|y,\theta_k)\right] ;$$

Maximization: Maximize $\theta \mapsto Q(\theta|\theta_k)$ in the feasible set Θ: Find $\theta_{k+1} \in \underset{\theta\in\Theta}{\operatorname{argmax}}\, Q(\theta|\theta_k)$.

We propose in this contribution a generalization of the SAEM algorithm, referred to as *approximated-SAEM*. Before describing our algorithm, we briefly review the SAEM algorithm in Section 2.2. Following Delyon et al. (1999), we restrict our attention to models for which the complete data likelihood belongs to the curved exponential family, which is a suitable framework in practice (Efron, 1978). We detail this setting in the following section.

2.1. Curved Exponential Family

The EM algorithm has been first introduced by Dempster et al. (1977) and convergence under very general conditions has been established by Wu (1983). Delyon et al. (1999) have reframed the convergence of the EM algorithm under easier hypotheses. We present here their results. In particular, they assume that the complete data likelihood belongs to the curved exponential family (M1).

(M1) The parameter space Θ is an open subset of $\mathbb{R}^{n_\theta}$. For all $y \in \mathcal{Y}$, $z \in \mathcal{Z}$ and $\theta \in \Theta$, the complete data likelihood function can be expressed as

$$q(y,z;\theta) \;=\; \exp\left(-\psi(\theta) + \langle\, S(y,z) \,|\, \phi(\theta)\,\rangle\right)$$

where $S : \mathbb{R}^{n_y} \times \mathbb{R}^{n_z} \to \mathcal{S} \subset \mathbb{R}^{n_s}$ is a Borel function, $\mathcal{S}$ is an open subset of $\mathbb{R}^{n_s}$, and where ϕ and ψ are functions of θ. The convex hull of $S(\mathbb{R}^{n_y} \times \mathbb{R}^{n_z})$ is included in $\mathcal{S}$. For all $\theta \in \Theta$, all $y \in \mathcal{Y}$, we have

$$\int_{\mathcal{Z}} \|S(y,z)\|\, q(z|y;\theta)\, \mathrm{d}\mu(z) \;<\; +\infty\,.$$

Let $\ell\colon \Theta \to \mathbb{R}$ and $L\colon \mathcal{S} \times \Theta \to \mathbb{R}$ defined as, for all $y \in \mathcal{Y}$,

$$\ell\colon \theta \mapsto \int_{\mathcal{Z}} q(y,z;\theta)\, \mathrm{d}\mu(z) \qquad \text{and} \qquad L\colon (s,\theta) \mapsto -\psi(\theta) + \langle\, s \,|\, \phi(\theta)\,\rangle\,.$$

Note that this definition implies that the function ℓ implicitly depends on y. However, as the observations y are assumed to be i.i.d, the counterpart function that would be defined on the whole set $\mathcal{Y}$ would be a sum over y of the defined ℓ. Additional hypotheses have to be set to ensure the regularity of the model.

(M2) The functions $\psi\colon \Theta \to \mathbb{R}$ and $\phi\colon \Theta \to \mathcal{S}$ are twice continuously differentiable on Θ;

(M3) The function $\bar{s} : \Theta \to \mathcal{S}$ is continuously differentiable on Θ, where $\bar{s}$ is defined as: For all $y \in \mathcal{Y}$,

$$\bar{s}\colon \theta \mapsto \int_{\mathcal{Z}} S(y,z) q(z|y;\theta)\, \mathrm{d}\mu(z) = \mathbb{E}_\theta\left[S(y,Z)\right] ;$$

(M4) The function $\ell\colon \Theta \to \mathbb{R}$ is continuously differentiable and for all $y \in \mathcal{Y}$ and $\theta \in \Theta$

$$\partial_\theta \int_{\mathcal{Z}} q(y,z;\theta)\, \mathrm{d}\mu(z) \;=\; \int_{\mathcal{Z}} \partial_\theta\, q(y,z;\theta)\, \mathrm{d}\mu(z)\,;$$

(M5) There exists a continuously differentiable function $\hat{\theta}\colon \mathcal{S} \to \Theta$ such that for all $\theta \in \Theta$ and $s \in \mathcal{S}$,

$$L(s, \hat{\theta}(s)) \;\geqslant\; L(s,\theta)\,.$$

For most models of practical interest (see for instance Section 4.2), the function $\theta \mapsto L(s;\theta)$ has a unique global *maximum* and the existence and the differentiability of $\hat{\theta}$ is a direct consequence of the implicit function theorem.

Let, for all $\theta \in \Theta$ and closed set $E \subset \Theta$, $d(\theta, E)$ denotes the distance of θ to E and, for all set E, clos(E) denotes the closure of E.

Theorem 1 (Convergence of the EM – Delyon et al. (1999))**.** *Assume that (M1 - 5) hold. Assume in addition that for any $\theta \in \Theta$, clos$(\mathcal{L}(\theta))$ is a compact subset of Θ. Then, for any initial point $\theta_0 = \theta$, the sequence $(\ell(\theta_k))_{k\in\mathbb{N}}$ is increasing and*

$$\lim_{k\to\infty} d(\theta_k, \mathcal{L}) = 0\,, \qquad \textit{where} \qquad \mathcal{L} = \{\theta \in \Theta | \partial_\theta \ell(\theta) = 0\}\,.$$

2.2. The Stochastic Approximation EM Algorithm

The basic idea ot the stochastic approximation version of the EM (SAEM) algorithm proposed by Delyon et al. (1999) is to split the expectation step into a simulation step and a stochastic averaging procedure. Starting from $Q_0(\theta) = 0$ for all $\theta \in \Theta$, we build an approximation of $\theta \mapsto Q(\theta|\theta_k)$ through stochastic approximation of the complete log-likelihood. We denote Q_k this approximation. As in the Monte-Carlo EM (MCEM) algorithm (Wei and Tanner, 1990; Fort and Moulines, 2003), the simulation step consists in generating realizations of the missing data vector under the posterior distribution $q(\cdot|y;\theta)$ and to use a Monte-Carlo approximation of the above expectation. Then, given an initial value θ_0, the SAEM algorithm updates θ_k into θ_{k+1} with the three following steps:

Simulation: Draw m_k samples $(z_{k[j]})_{j\in[\![1,m_k]\!]}$ of the latent variables z_k under the posterior density $q(\,\cdot\,|y;\theta_k)$;

Stochastic Approximation: Update $Q_k(\theta)$ according to

$$Q_{k+1}(\theta) \;=\; Q_k(\theta) + \gamma_k\left(\frac{1}{m_k}\sum_{j=1}^{m_k} \log q(y, z_{k[j]};\theta) - Q_k(\theta)\right),$$

where $(\gamma_k)_{k\in\mathbb{N}}$ is a sequence of positive step-size.

Maximization: Maximize Q_{k+1} in the feasible set Θ, *i.e.* find $\theta_{k+1} \in \operatorname*{argmax}_{\theta\in\Theta} Q_{k+1}(\theta)$.

Once the step size γ_k decreases, we can consider a constant number of simulations (Delyon et al., 1999). In practice, as the simulation step is generally the most computationally costly, we use a single simulation. Moreover, for curved exponential families, the stochastic approximation step is more conveniently (and equivalently) replaced by an update of the estimation of the conditional expectation of the sufficient statistics. Let s_k denote the k^{th} approximation of the sufficient statistics. Then, the k-th iteration of the SAEM summarizes in:

$$s_{k+1} \;=\; s_k + \gamma_k\big(S(y, z_k) - s_k\big) \qquad \text{and} \qquad \theta_{k+1} \;=\; \hat{\theta}(s_{k+1})\,, \qquad \text{where} \qquad z_k \;\sim\; q(\,\cdot\,|y;\theta_k)\,,$$

and where s_k is initialized to zero: $s_0(\theta) = 0$ for all $\theta \in \Theta$.

2.2.1. Convergence of the SAEM Algorithm

Convergence of the SAEM algorithms toward a local *maximum* of the likelihood is ensured under general conditions (Delyon et al., 1999). Before establishing the convergence of the approximated-SAEM introduced in Section 3, we recall here the result obtained by Delyon et al. (1999).

Let $\mathcal{F} = \{\mathcal{F}_k\}_{k\in\mathbb{N}}$ the natural filtration with respect to the process $(z_k)_{k\in\mathbb{N}}$, namely

$$\mathcal{F}_k \;=\; \sigma\left(\{z_1, z_2, \ldots, z_k\}\right) = \sigma(\{z_1^{-1}(A), z_2^{-1}(A), \ldots z^{-1}(A) | A \in \mathcal{B}(\mathcal{Z})\})\,.$$

(SAEM1) For all $k \in \mathbb{N}$, $\gamma_k \in [0,1]$, $\sum_{k=1}^{\infty} \gamma_k = \infty$ and $\sum_{k=1}^{\infty} \gamma_k^2 < \infty$;

(SAEM2) The functions $\psi \colon \Theta \to \mathbb{R}$ and $\phi \colon \Theta \to \mathcal{S}$ are m times differentiable;

(SAEM3) For all positive Borel functions ϕ, for all $k \in \mathbb{N}$ and all $y \in \mathcal{Y}$,

$$\mathbb{E}\big[\phi(Z_{k+1})\big|\mathcal{F}_k\big] \;=\; \int_{\mathcal{Z}} \phi(z) q(z|y;\theta_k)\,\mathrm{d}\mu(z)\,;$$

(SAEM4) For all $\theta \in \Theta$, all $y \in \mathcal{Y}$ and all $k \in \mathbb{N}$,

$$\int_{\mathcal{Z}} \|S(y,z)\|^2\, q(y,z;\theta)\,\mathrm{d}\mu(z) \;<\; +\infty\,.$$

The proof of this theorem relies on convergence results for Robbins–Monro stochastic approximation procedure Robbins and Monro (1951). The assumption (SAEM1) is characteristic of this type of procedures in which the step-size has to decrease not too fast.

Theorem 2 (Convergence of the SAEM – Delyon et al. (1999))**.** *Assume that (M1 - 5) and (SAEM1 - 4) hold. Assume in addition that, with probability 1, clos*$(\{s_k\}_{k\in\mathbb{N}^*})$ *is a compact subset of* $\mathcal{S}$*. Then, with probability 1,*

$$\lim_{k\to\infty} d(\theta_k, \mathcal{L}) = 0\,, \qquad \textit{where} \qquad \mathcal{L} = \{\theta \in \Theta | \partial_\theta \ell(\theta) = 0\}\,.$$

This theorem ensures the convergence of the SAEM algorithm only toward a stationary point of the log-likelihood. To ensure the convergence of the algorithm toward a local *maximum* we have to assume at least local convexity of the log-likelihood. Some conditions upon which the convergence toward local *maxima* is guaranteed are given in Section 7 of Delyon et al. (1999). As these conditions are model-dependent, we do not focus on this aspect in this contribution. Basically, this is equivalent to assuming that the observed likelihood is convex.

3. A New Stochastic Approximation Version of the EM Algorithm

Despite appealing features, sampling from the posterior distribution may be intractable or have a high computational cost. Moreover, the limit position of the SAEM algorithm can strongly depend on its starting position θ_0. To cope with this two issues, we propose in this contribution a new stochastic approximation version of the EM, referred to as approximated-SAEM, in which we do not sample from the exact distribution in the expectation phase of the procedure.

Similar to the standard SAEM, our version is to replace the exact computation of the expectation by a stochastic approximation of the latter. In the original SAEM, the simulation consists of generating realizations of the missing data vector under the posterior distribution $q(\cdot|y;\theta)$. Here, we propose to sample under *approximation* of the posterior distribution.

Let $(\gamma_k)_{k\in\mathbb{N}}$ be a sequence of positive step-size for the stochastic approximation, and $(\tilde{q}_k)_{k\in\mathbb{N}}$ be a sequence of *approximated* distributions on $\mathcal{Z}\times\Theta$ such that for all $k \in \mathbb{N}$ and all $\theta \in \Theta$, $\tilde{q}_k(\cdot;\theta)$ is integrable on $\mathcal{Z}$ with respect to the measure μ. As in the SAEM, once the step size γ_k decreases, we can set the number of simulations to one. In other words, our version still requires only one realization of the missing data for each iteration, which allows an efficient control of the computation time.

The approximated-SAEM iterates the following three steps:

Simulation: Sample the latent variable $\tilde{z}_k$ under the approximated density $\tilde{q}_k(\cdot;\theta_k)$;

Stochastic Approximation: Update $Q_k(\theta)$ as $Q_{k+1}(\theta) = Q_k(\theta) + \gamma_k\big(\log q(y,\tilde{z}_k;\theta) - Q_k(\theta)\big)$;

Maximization: Maximize Q_{k+1} in the feasible set Θ, *i.e.* find $\theta_{k+1} \in \operatorname*{argmax}_{\theta\in\Theta} Q_{k+1}(\theta)$.

The detailed constrains of approximated densities are given in Assumption (A) of Theorem 3. Note already that without approximation, *i.e.* if the approximated densities $\tilde{q}_k$ match with the correct posterior distribution, we feature the classical SAEM. Moreover, the approximated densities $\tilde{q}_k$ may not depend on the observations y, as in variational Bayesian methods or may be done by ABC samplers as in ABC-SAEM (Picchini and Samson, 2018). In Section 3.2, we propose a way to build a sequence $(\tilde{q}_k)_{k\in\mathbb{N}}$ leading to good properties in practice; theoretical guarantees are given in the following section.

3.1. Convergence of the Approximated-SAEM Algorithm

As said previously, we restrict our attention to models for which the complete data likelihood belongs to the curved exponential family. In this way, as for the SAEM algorithm, the stochastic approximation can be held on the sufficient statistic S. Then, the k-th iteration of the approximated-SAEM summarizes in:

$$s_k \;=\; s_{k-1} + \gamma_k\big(S(y,\tilde{z}_k) - s_{k-1}\big) \qquad \text{and} \qquad \theta_k \;=\; \hat{\theta}(s_k)\,, \qquad \text{where} \qquad \tilde{z}_k \;\sim\; \tilde{q}_k(\cdot;\theta_{k-1})\,, \tag{1}$$

and where s_k is initialized to zero: $s_0(\theta) = 0$ for all $\theta \in \Theta$.

Let consider the following assumptions which are generalization of the ones of Delyon et al. (1999): In Theorem 3, a hypothesis stated with a $(^\star)$ means that it is a direct generalization of the corresponding one from Section 2; on the contrary, hypotheses stated without are unchanged compared to the original one.

We keep the notations of Section 2. Let $\widetilde{\mathcal{F}} = \{\widetilde{\mathcal{F}}_k\}_{k\in\mathbb{N}}$ the natural filtration with respect to the process $(\tilde{z}_k)_{k\in\mathbb{N}}$.

(M1*) The parameter space Θ is an open subset of $\mathbb{R}^{n_\theta}$. For all $y \in \mathcal{Y}$, $z \in \mathcal{Z}$ and $\theta \in \Theta$, the complete data likelihood function can be expressed as

$$q(y,z;\theta) \;=\; \exp\big(-\psi(\theta) + \langle\, S(y,z)\,|\,\phi(\theta)\,\rangle\big)$$

where $S : \mathbb{R}^{n_y} \times \mathbb{R}^{n_z} \to \mathcal{S} \subset \mathbb{R}^{n_s}$ is a Borel function and $\mathcal{S}$ is an open subset of $\mathbb{R}^{n_s}$. The convex hull of $S(\mathbb{R}^{n_y} \times \mathbb{R}^{n_z})$ is included in $\mathcal{S}$. For all $\theta \in \Theta$, all $y \in \mathcal{Y}$ and all $k \in \mathbb{N}$, we have

$$\int_{\mathcal{Z}} \|S(y,z)\|\, \tilde{q}_k(z;\theta)\,\mathrm{d}\mu(z) < +\infty \qquad \text{and} \qquad \int_{\mathcal{Z}} \|S(y,z)\|\, q(z|y;\theta)\,\mathrm{d}\mu(z) < +\infty\,.$$

Hypothesis (M1*) differs from (M1) as we do not only require the function $z \mapsto \|S(y,z)\|$ to be integrable with respect to the posterior measure $q(\cdot|y;\theta)\,\mathrm{d}\mu$, but also with respect to all approximated distributions $\tilde{q}_k(\cdot;\theta)\,\mathrm{d}\mu$, for all parameters $\theta \in \Theta$, all observations $y \in \mathcal{Y}$ and all iterations $k \in \mathbb{N}$.

(SAEM3*) For all positive Borel functions ϕ, for all $k \in \mathbb{N}$ and all $y \in \mathcal{Y}$,

$$\mathbb{E}\big[\phi(Z_{k+1})\big|\widetilde{\mathcal{F}}_k\big] \;=\; \int_{\mathcal{Z}} \phi(z)\tilde{q}_k(z;\theta_k)\,\mathrm{d}\mu(z) \qquad \text{and} \qquad \mathbb{E}\big[\phi(Z_{k+1})\big|\mathcal{F}_k\big] \;=\; \int_{\mathcal{Z}} \phi(z) q_k(z|y;\theta_k)\,\mathrm{d}\mu(z)\,;$$

(SAEM4*) For all $\theta \in \Theta$, all $y \in \mathcal{Y}$ and all $k \in \mathbb{N}$,

$$\int_{\mathcal{Z}} \|S(y,z)\|^2\, \tilde{q}_k(z;\theta)\,\mathrm{d}\mu(z) < +\infty\,.$$

Likewise, (SAEM3*) is similar to (SAEM3), except that we assume that, given $\theta_0,\dots,\theta_k$, *both* simulated latent variables $\tilde{z}_1,\dots,\tilde{z}_k$ and $z_1,\dots,z_k$ are conditionally independent, given their respective natural filtration. In Assumption (SAEM4*), we demand the integrability of $z \mapsto \|S(y,z)\|^2$ with respect to the measures $\tilde{q}_k(z;\theta)\,\mathrm{d}\mu$.

The following theorem ensures the convergence of our new stochastic approximation version of the EM algorithm. This theorem is the approximated counterpart of Theorem 2 of Delyon et al. (1999).

Theorem 3 (Convergence of the approximated-SAEM). *Assume that (M1⋆), (M2 - 5), (SAEM1 - 2) and (SAEM3⋆ - 4⋆) hold. Assume in addition that:*

(A) *For all $y \in \mathcal{Y}$, the sequence $\left(\tilde{q}_k(\cdot;\theta)\right)_{k\in\mathbb{N}}$ converges in mean on every compact subset of Θ for the measure $S.\mu$ to $q(\cdot|y;\theta)$, that is to say for all observations $y \in \mathcal{Y}$ and all compact $\mathcal{K} \subset \Theta$,*

$$\lim_{k\to\infty} \left\{ \sup_{\theta\in\mathcal{K}} \int_{\mathcal{Z}} S(y,z)\left(\tilde{q}_k(z;\theta) - q(z|y;\theta)\right) \mathrm{d}\mu(z) \right\} = 0\,;$$

(B) *With probability 1, $clos\left(\{s_k\}_{k\in\mathbb{N}^*}\right)$ is a compact subset of $\mathcal{S}$.*

Let $\mathcal{L} = \{\theta \in \Theta | \partial_\theta \ell(\theta) = 0\}$. Then, with probability 1,

$$\lim_{k\to\infty} d(\theta_k, \mathcal{L}) = 0\,.$$

Hypothesis (A) makes explicit what we mean by a sequence of approximated densities. In particular, it allows a wide variety of numerical schemes; we propose an example of practical interest in Section 3.2. Note that (SAEM4⋆) and (A) ensure the function $z \mapsto \|S(y,z)\|^2$ to be integrable with respect to the measure $q(y,z\theta)\,\mathrm{d}\mu$. The compactness assumption (B) ensures asymptotic control in space of the sequence produced by the algorithm. This assumption was already required by Theorems 1 and 2. In Section 3.1.1, we present a way to overcome it through a stabilization procedure.

Note that the statement of Theorem 3 is very similar to the corresponding one of Delyon et al. (1999), namely Theorem 2 which establish the convergence of the SAEM. In other words, approximate the posterior distribution in the simulation step does not require supplementary considerations to still guarantee the convergence of the sequence $(\theta_k)_{k\in\mathbb{N}}$. Thus, the scope of application of the approximated-SAEM algorithm is at least as unrestrictive as the one of the SAEM.

The demonstration consists in applying Theorem 2 of Delyon et al. (1999) ; we recall it in Appendix A (Theorem 4). In particular, (SA0 - 4) refer to their hypotheses. Moreover, since it is used in our demonstration, we also recall Lemma 2 from the same paper (Lemma 2 in Appendix A). For sake of simplicity, we prove the convergence of the approximated-SAEM under the compactness condition (B). However, the result remains true even if (B) is not satisfied, on condition of having recourse to the truncation on random boundaries procedure described hereafter (Algorithm 1).

Proof. As for all $k \in \mathbb{N}$, $\gamma_k \in [0,1]$, (SA0) is verified under (M1⋆) and (SAEM1). Moreover, (SA1) is implied by (SAEM1) and (SA3) by (B). Note that under Assumption (B), there exists, with probability 1, a compact set K such that for all $k \in \mathbb{N}$, $s_k \in K$.

Let, for all $s \in \mathcal{S}$ and $k \in \mathbb{N}$, $h(s) = \bar{s}(\hat{\theta}(s)) - s$,

$$e_k \;=\; S(y,\tilde{z}_k) - \mathbb{E}\left[S(y,\tilde{z}_k)|\widetilde{\mathcal{F}}_{k-1}\right] \qquad \text{and} \qquad r_k \;=\; \mathbb{E}\left[S(y,\tilde{z}_k)|\widetilde{\mathcal{F}}_{k-1}\right] - \bar{s}(\hat{\theta}(s_{k-1}))$$

such that Equation (1) writes on Robbins-Monro type approximation procedure.

As Lemma 2 depends only of the meanfield of the model, it can be applied as it is. Thus, (SA2.i) is satisfied with the Lyapunov function $V = -\ell \circ \hat{\theta}$,

$$\{s \in \mathcal{S} | F(s) = 0\} \;=\; \{s \in \mathcal{S} | \partial_s V(s) = 0\} \qquad \text{and} \qquad \hat{\theta}\left(\{s \in \mathcal{S} | F(s) = 0\}\right) \;=\; \{\theta \in \Theta | \partial_\theta \ell(\theta) = 0\} \;=\; \mathcal{L}\,.$$

Moreover, (SA2.ii) is satisfied due to the Sard theorem and (SAEM2). We only need to focus on (SA4).

Set for all $n \in \mathbb{N}^*$, $E_n = \sum_{k=1}^n \gamma_k e_k$. The sequence $(E_n)_{n\in\mathbb{N}^*}$ is a $\widetilde{\mathcal{F}}$-martingale: for all $m > n$, $\mathbb{E}\left[E_m|\widetilde{\mathcal{F}}_n\right] = E_n$ as for all $k > n$, $\widetilde{\mathcal{F}}_n \subset \widetilde{\mathcal{F}}_{k-1}$. Moreover, for all $n \in \mathbb{N}$,

$$\mathbb{E}\left[\left\|S(y,\tilde{z}_{n+1}) - \mathbb{E}\left[S(y,\tilde{z}_{n+1})|\widetilde{\mathcal{F}}_{n+1}\right]\right\|^2 \middle| \widetilde{\mathcal{F}}_n\right] \leqslant \mathbb{E}\left[\|S(y,\tilde{z}_{n+1})\|^2 \middle| \widetilde{\mathcal{F}}_n\right] < \infty \quad \text{a.s.}$$

since by (B) and (M5), with probability 1, $\hat{\theta}(s_n)$ is in the compact set $\hat{\theta}(K) \subset \Theta$. So,

$$\sum_{n=1}^{\infty} \mathbb{E}\left[\|E_{n+1} - E_n\|^2 \,\middle|\, \widetilde{\mathcal{F}}_n\right] \leqslant \sum_{n=1}^{\infty} \gamma_{n+1}^2 \,\mathbb{E}\left[\|S(y, \tilde{z}_{n+1})\|^2 \,\middle|\, \widetilde{\mathcal{F}}_n\right] < \infty \quad \text{a.s.}\,.$$

According to Theorem 2.15 of Hall and Heyde (1980), with probability 1, $\lim_{n\to\infty} E_n$ exists. Moreover,

$$r_n \;=\; \int_Z S(y,z)\left(q(z|y,\hat{\theta}(s_{n-1})) - \tilde{q}_n(z,\hat{\theta}(s_{n-1}))\right)\mathrm{d}\mu(z)$$

for all $n \in \mathbb{N}$, which converge to 0 according to hypothesis (A), proving (SA4).

Thus, Theorem 4 applies and almost surely

$$\limsup_{k\to\infty} d(s_k, \{s \in \mathcal{S} | \partial_s V(s) = 0\}) \;=\; \limsup_{k\to\infty} d(s_k, \{s \in \mathcal{S} | F(s) = 0\}) = 0\,.$$

Lastly, by continuity of $\hat{\theta}\colon \mathcal{S} \to \Theta$,

$$\limsup_{k\to\infty} d\left(\hat{\theta}(s_k), \hat{\theta}\big(\{s \in \mathcal{S} | F(s) = 0\}\big)\right) \;=\; \limsup_{k\to\infty} d(\theta_k, \mathcal{L}) = 0\,.$$

□

The obtained results demonstrate that, under appropriate conditions, the sequence $(\theta_k)_{k\in\mathbb{N}}$ converges to a connected component of the set $\mathcal{L}$ of stationary points of ℓ. Moreover, some conditions upon which the convergence toward local *maxima* is guaranteed are given in Section 7 of Delyon et al. (1999). As these conditions only depend on the design of the model and not on the definition of the optimizing sequence $(\theta_k)_{k\in\mathbb{N}}$, the corresponding theorems remain exact in our context leading to classical hypotheses ensuring convergence toward local *maxima*.

3.1.1. Truncation on Random Boundaries

As said and illustrated in the demonstration, the compactness condition (B) makes it possible to control asymptotically the sequence of stochastic approximations $(s_k)_{k\in\mathbb{N}}$. However, in practice, checking this condition may be intractable. In that case, we have to recourse to a stabilization procedure. We proceed as in (Andrieu et al., 2006). Let $(\mathcal{K}_n)_{n\in\mathbb{N}}$ be an exhaustion by compact sets of the space $\mathcal{S}$, *i.e.* be a sequence of compact subsets of $\mathcal{S}$ such that

$$\bigcup_{n\in\mathbb{N}} \mathcal{K}_n = \mathcal{S} \qquad \text{and} \qquad \forall k \in \mathbb{N}, \quad \mathcal{K}_n \subset \mathrm{int}(\mathcal{K}_{n+1})\,,$$

where $\mathrm{int}(A)$ denotes the interior of the set A. The main idea is to reset the sequence s_k to an arbitrary point every time s_k wanders out of the compact subset $\mathcal{K}_{n_k}$, where n_k is the number of projections up to the k-th iteration. Let $(\varepsilon_k)_{k\in\mathbb{N}}$ be a monotone non-increasing sequence of positive numbers and let K be a subset of $\mathcal{Z}$. Last, let $\Pi\colon \mathcal{Z} \times \mathcal{S} \to K \times \mathcal{K}_0$ be a measurable function (See (Andrieu et al., 2006) for details about the way to choose Π). The stochastic approximation with truncation on random boundaries is summarized in Algorithm 1.

3.2. A Tempering Version of the SAEM

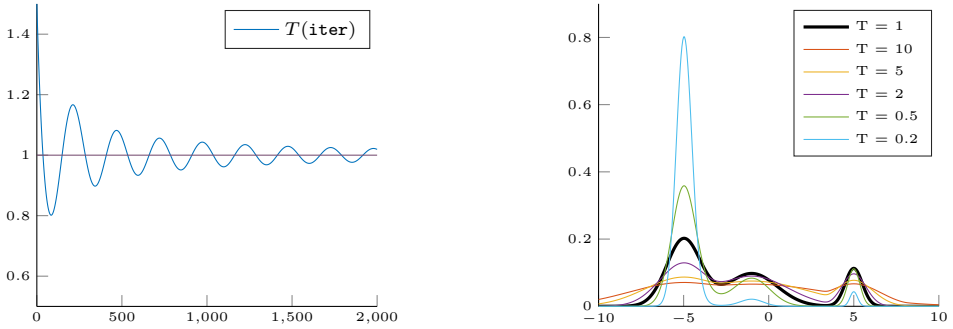
We focus in the following on an instantiation of the approximated-SAEM, leading to the *tempering-SAEM*. Let $(T_k)_{k\in\mathbb{N}}$ be a sequence of positive numbers such that $\lim_{k\to\infty} T_k = 1$. We set, for all $y \in \mathcal{Y}$, all $z \in \mathcal{Z}$, all $\theta \in \Theta$ and all $k \in \mathbb{N}$,

$$\tilde{q}_k(z;\theta) \;=\; \frac{1}{c_\theta(T_k)}\, q(z|y;\theta)^{1/T_k}\,,$$

where $c_\theta(T_k)$ is a scaling constant.

Algorithm 1: Stochastic approximation with truncation on random boundaries

Set $n_0 = 0$, $s_0 \in \mathcal{K}_0$ and $\tilde{z}_0 \in K$
for *all* $k \in \mathbb{N}$ **do**
  Sample $\tilde{z}^* \sim \tilde{q}_k(\cdot; \theta_{k-1})$
  Compute $s^* = s_{k-1} + \gamma_k\big(S(y, \tilde{z}^*) - s_{k-1}\big)$
  if $s^* \in \mathcal{K}_{n_{k-1}}$ **then**
    Set $(\tilde{z}_k, s_k) = (\tilde{z}^*, s^*)$
  else
    Set $(\tilde{z}_k, s_k) = \Pi(\tilde{z}_{k-1}, s_{k-1})$ and $n_k = n_{k-1} + 1$
  end
  Set $\theta_k = \hat{\theta}(s_k)$
end



(a) Tempering scheme (b) Tempering distributions

Figure 1: *Construction of the temperature scheme.* Fig. 1a: Evolution of the temperature over iteration for the tempering-SAEM. Fig. 1b: Influence of the temperature over the pattern of the distribution.

Let $y \in \mathcal{Y}$ and $\mathcal{K} \subset \Theta$ compact. Then, by continuity of the function $\theta \mapsto q(z|y;\theta)$, it exists $M \in \mathbb{R}$ such that

$$\sup_{\theta \in \mathcal{K}} |S(y,z)\left(\tilde{q}_k(z;\theta) - q(z|y;\theta)\right)| \leqslant \sup_{\theta \in \mathcal{K}} M \left| 1 - \frac{1}{c_\theta(T_k)} \exp\left(-\Big(1 - \frac{1}{T_k}\Big) q(z|y,\theta_k)\right)\right| .$$

Thus, as $\mathcal{K}$ is compact, (A) is satisfied.

Note that our tempering-SAEM differs from the simulated annealing version of Lavielle and Moulines (1997) as we do not modify the model but only the sampling-step of the estimation algorithm.

3.2.1. Escape Local Maxima

This scheme has been built with the intuition of the simulated annealing (Kirkpatrick et al., 1983): the sequence $(T_k)_{k\in\mathbb{N}}$ has to be interpreted as a sequence of temperatures. The higher T_k, the more the corresponding distribution $\tilde{q}_k$ lies flat and the (approximated) hidden variable z_k can explore all the set $\mathcal{Z}$. On the contrary, a low temperature will freeze the exploration of z_k (see Figure 1b). Thus, finding an appropriate sequence $(T_k)_{k\in\mathbb{N}}$ to keep a balance between both behaviors is a great methodological challenge.

We propose here an oscillatory tempering pattern which oscillates around one with decreasing amplitude. In other words given the decreasing and amplitude rate a and b, the scaling-parameter r and the delay c, we define our sequence of temperatures by: for all $k \in \mathbb{N}$,

$$T_k \;=\; 1 + a^\kappa + b\,\frac{\sin(\kappa)}{\kappa}\,, \qquad \text{where} \qquad \kappa \;=\; \frac{k + c \times r}{r}\,.$$

We design this scheme to decrease, with an exponential rate toward 1, with dampened oscillations. In this form, the tempering scheme includes the tempering scheme used by MONOLIX. Even so, the experiments conducted in section 4 tend to show that the exponential decrease is not necessary. In particular, a is set to zero for all the experiments. However, in order to encompass the tempering scheme implemented in MONOLIX, we have chosen to keep this parameter in the generic form.

In order for the tempering scheme to converge toward 1, simply require the exponential rate a to be in $[0, 1[$. In particular, the parameter b can be chosen independently negative or positive. A positive b will flatten the distribution at the beginning of the optimization procedure. On the contrary, a negative b will make the profile of the distribution more prickly. It can be interesting to enforce the distinction of two close modes, as in Section 4.1.2.

Due to the oscillations of the temperature, the latent variable z_k will explore and gather in turns. Thus, in the case of multimodal density, the latent variable will be able to switch from one mode to another during the heating steps and to explore these same modes during the cooling phases. In particular, during the optimization, the tempering-SAEM may escape from local minima in which the SAEM would get stuck. Figures B.15 and B.16 effectively illustrate this phenomenon. In this way, the local *maxima* of the likelihood can be avoided. Moreover, as the approximated distributions regularly gather around the modes of the posterior distribution $q(\cdot|y;\theta_k)$, the exploration of z will stabilize and the algorithm will converge.

Although the analysis of this algorithm is heuristic, the simulations (see the following section) confirm the intuition and give good results. A theoretical analysis is an ongoing problem.

4. Application and Experiments

In this section, we will focus on escaping local *maxima* problem. Concerning intractable sampling and a way to built approximations in this context, we may refer to the convincing results obtained by Picchini and Samson (2018) with the ABC-SAEM for instance.

As explained in the previous paragraph, the tempering-SAEM allows us to escape from local *maxima*. To illustrate this phenomenon, we propose two applications: cluster analysis through Gaussian mixture model (Melnykov and Maitra, 2010) and independent factor analysis which can lead to blind source separation (Moulines et al., 1997; Attias, 1999; Allassonnière and Younes, 2012).

4.1. Multivariate Gaussian Mixture Models

Before considering a more realistic application, we first present an application of the tempering-SAEM to multivariate Gaussian mixture model (GMM). Actually, in spite of an apparent simplicity, this model illustrates well the main features of our algorithm.

Let $y = (y_i)_{i\in[\![1,n]\!]} \in \mathbb{R}^{nd}$ be a n-sample of $\mathbb{R}^d$. We assume that y is distributed under a a weighted sum of m d-dimensional Gaussians: Given the weights $\alpha = (\alpha_j)_{j\in[\![1,m]\!]} \in [0,1]^m$ such that $\sum_{j=1}^m \alpha_j = 1$, the centroids $\mu = (\mu_j)_{j\in[\![1,m]\!]} \in \mathbb{R}^{md}$ and the covariance matrices $\Sigma = (\Sigma_j)_{j\in[\![1,m]\!]} \in (\mathcal{S}_d\mathbb{R})^m$, we assume that

$$y|z,\theta \sim \bigotimes_{i=1}^{n} \mathcal{N}(\mu_{z_i}, \Sigma_{z_i}) \qquad \text{and} \qquad z|\theta \sim \sum_{j=1}^{m} \alpha_j \delta_j \,,$$

where $\theta = (\alpha, \mu, \Sigma)$ and $z = (z_i)_{i\in[\![1,n]\!]}$ is the latent variable specifying the identity of the mixture component of each observation. In the following, we compare the efficiency of the EM, the SAEM and the tempering-SAEM algorithms to produce a *maximum* likelihood estimate of the parameters with the *a priori* given exact number of components m.

Classically, as closed-form expressions are possible for finite GMM, the EM algorithm is a very popular technique used to produce the *maximum* likelihood estimation of the parameters (Xu and Jordan, 1996; McLachlan and Peel, 2000). However, the computational cost can be prohibitive. A faster procedure is to use the SAEM algorithm. Nevertheless, both algorithms are very sensitive to its initial position: solutions can highly depend on their starting point and consequently produce sub-optimal *maximum* likelihood estimates

(Biernacki et al., 2003). The tempering-SAEM appears as a way to escape from local *maxima* and reach global *maxima* more often.

Note that the GMM does not satisfy the compactness hypothesis (B) (Titterington et al., 1985). We then need to use the truncation on random boundaries described in Section 3.1.1. In practice, as long as the sequence of estimates remains within a compact, the convergence of the three algorithms is ensured. The three procedures are detailed in Appendix B.

4.1.1. Insensitivity of the tempering-SAEM to Initialization

To estimate the sensitivity of the tempering-SAEM algorithm to its initial position, we generate a synthetic dataset (Figure 2) and perform the estimation 500 times for the three algorithms, with the same sequences of points chosen at random within the dataset.

For the sake of brevity, we will refer to the Kullback-Leibler divergence between two centered Gaussian distributions as the Kullback-Leibler divergence between their corresponding covariance matrices. The relative errors for α and μ and the Kullback-Leibler divergence between the true covariance matrices Σ and the estimated one are compiled in Figures 3b, 3d and 3f. The class refers to the ones in Figure 2. We consider the algebraic relative error for α so that we can deduce if the studied algorithm tends to empty (Class E) or overfill (Class B) the classes. First, the tempering-SAEM is always competitive with the EM and the SAEM and most of the time greater. In other words, the global *maximum* is more often reached while tempering the posterior distribution. Moreover, while EM and SAEM achieve fairly identical results, the tempering-SAEM can discriminate overlapped classes. Figures 3a, 3c and 3e displays the result of a type run for each of the three algorithms, with the same initial points (the blue crosses). Class A, which is the only isolated class, is seemingly the best learned. The EM and SAEM seem to empty Class C for the benefit of Class B and merge them on a "super-class" as if there were only 5 components in the Gaussian mixture.

4.1.2. Escaping Local Maxima

We then consider a situation known to be badly managed by the EM algorithm. Namely, we consider a three clusters' dataset. One cluster is on the far right side, two are on the far left side and all the three clusters are equiprobable. Moreover, we want to study the influence of the distance between the two left clusters. So we build three datasets: one where the two left clusters are properly distinct, one where they are close and a last one where they are almost merged. The three datasets are displayed at Figure 4. For each dataset, we perform the optimization for two different initial positions, referred as initialization 1 and 2 in the following. In the first case, the three centroïdes μ are initialized at the barycenter of the observed data. In the second one, we initialize two means on the right side and one on the left side.

For each situation, we perform the estimation through all the three algorithms. We present at Tables 1, 2 and 3 the relative errors for the different parameters. As previously, we consider the algebraic relative error for the weights α. To better understand the behavior of the different algorithms, we also provide a box plot of these relative errors in Figures 5a-b, 6a-b and 7a-b. The SAEM algorithm tends to empty classes for the benefit of other(s). It seems to be less the case for the tempering-SAEM. Note that, whatever the dataset, if the mean parameters are initialized to the mean of the dataset, the EM algorithm does not move. Thus, the error concerning the mixture proportion α seems to be very small, but this is only due to the initialization of the parameters α in favor of an equiprobable mixture.

Figures 5c-d, 6c-d and 7c-d display the mean of the estimated $\widehat{\mu}$ and $\widehat{\Sigma}$ by the three algorithms (EM *versus* SAEM *versus* tempering-SAEM), for each dataset and each initial position.

The tempering-SAEM succeed to accurately estimate all the parameters related to the first and second datasets (Figures 5 and 6; Tables 1 and 2), even when two of the mean parameters μ are initialized in the right cluster (Initialization 2). When the parameters μ are initialized to the barycenter of the dataset, the tempering-SAEM still accurately estimates the different parameters, including for the dataset III where the left clusters are merged (Figure 7c). However, when two of the mean parameters are initialized within the single right cluster, the tempering-SAEM does not succeed to capture the two left classes if they are to close (Figure 7d), but this can easily be explained by the distribution of the observations (Figure 4c).

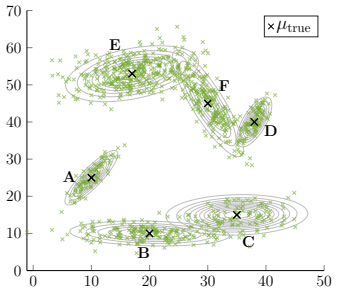
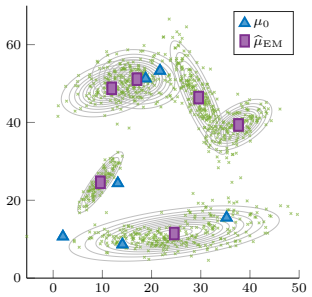
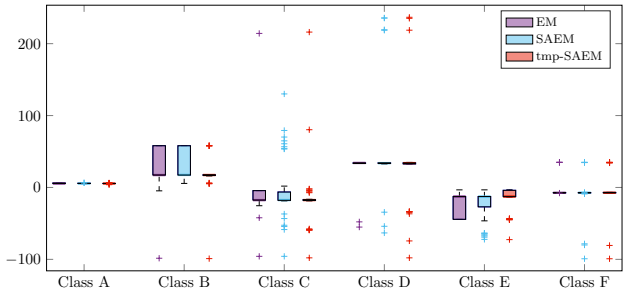


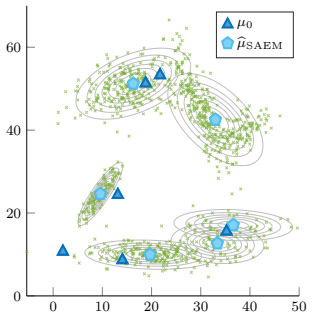
Figure 2: *Learning dataset* used to perform the experience regarding Section 4.1.1.



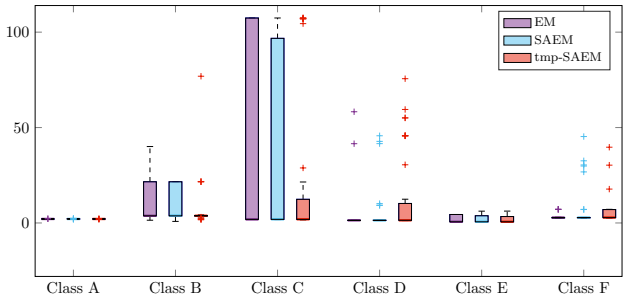
(a) EM Algorithm



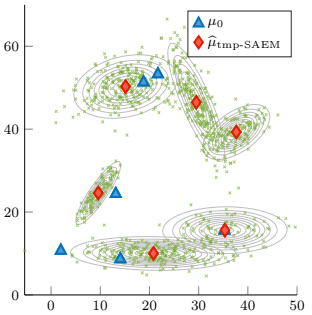
(b) Relative error (expressed as a percentage) for α



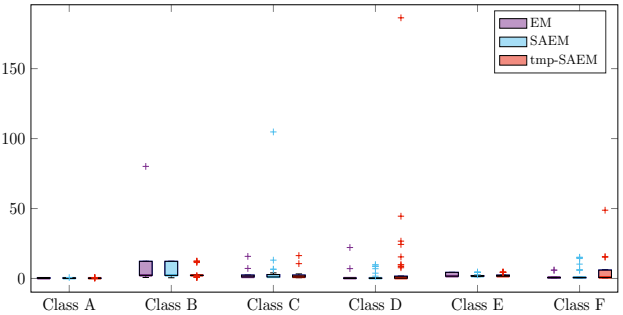
(c) SAEM Algorithm



(d) Relative error (expressed as a percentage) for μ

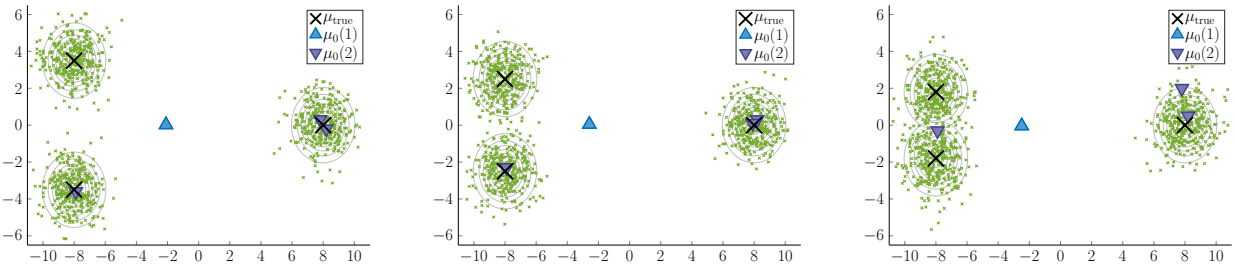


(e) tempering-SAEM Algorithm



(f) KL distance between the true Σ and the estimated ones

Figure 3: *Performance of the estimation for the Multivariate Gaussian mixture model.* Figs. 3a, 3c and 3e: Qualitative comparison of the *maximum* likelihood estimation of the parameters. The estimation is performed with the same initial points (the blue crosses). Figs. 3b, 3d and 3f: Relative error (expressed as a percentage) for the weights α and the centroïds μ. Kullback-Leibler distance between the true covariance matrices Σ and the estimated ones, for 500 runs and $n = 1000$.

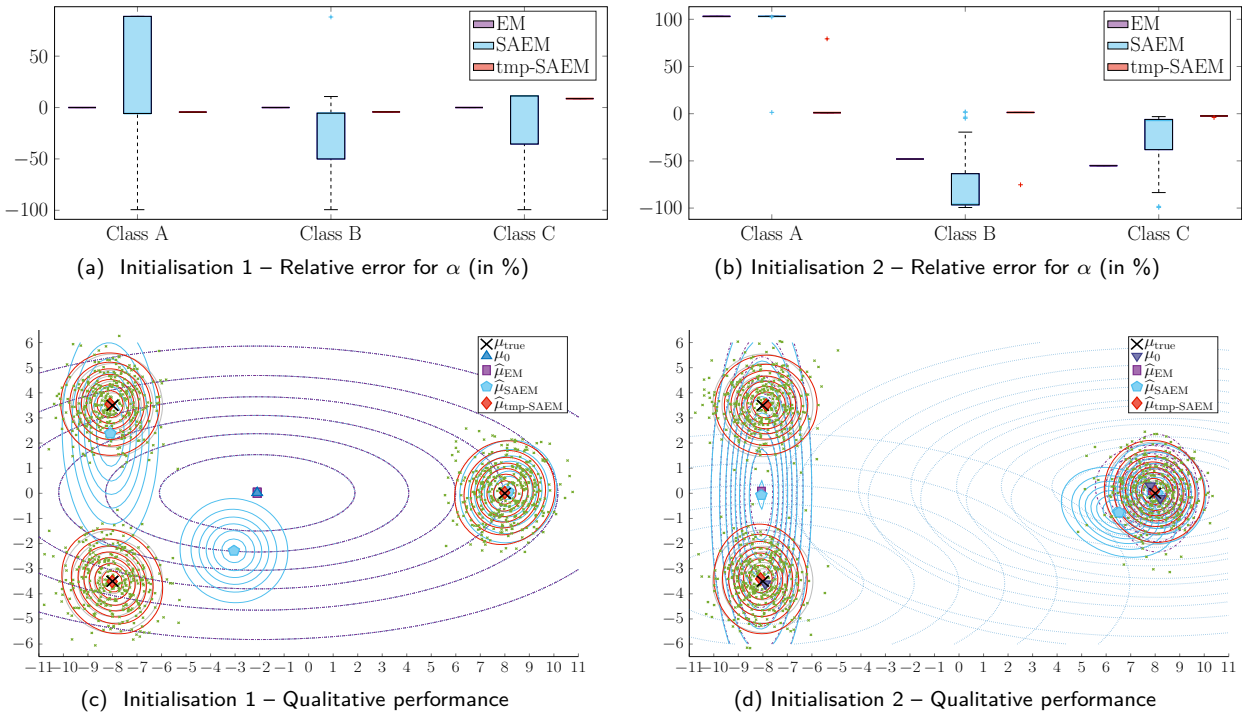


(a) Dataset I (b) Dataset II (c) Dataset III

Figure 4: *The three scattered clusters' datasets* used to perform the experiences regarding Section 4.1.2. For each dataset, we consider two possible initial positions for the means μ: either all at the barycenter of the dataset (the blue asterisk) or two of them in the single right cluster and the last mean on the left side (the orange asterisks).

	EM - 1	**EM** - 2	**SAEM** - 1	**SAEM** - 2	**tmp-SAEM** - 1	**tmp-SAEM** - 2
$\widehat{\alpha}_A$	***0.00***	103.10	24.83 (46.24)	99.46 (18.99)	−4.46 (0.00)	***2.01 (8.54)***
$\widehat{\alpha}_B$	***0.00***	−48.02	−19.41 (25.89)	−77.72 (28.72)	−4.23 (0.00)	***0.39 (8.35)***
$\widehat{\alpha}_C$	***0.00***	−55.08	−5.42 (26.22)	−21.87 (24.94)	8.69 (0.00)	***-2.40 (0.18)***
$\widehat{\mu}_A$	78.46	39.44	14.28 (18.28)	38.07 (7.15)	***1.24 (0.00)***	***1.62 (4.18)***
$\widehat{\mu}_B$	78.46	185.93	58.54 (84.14)	168.49 (37.28)	***0.17 (0.00)***	***2.56 (17.03)***
$\widehat{\mu}_C$	126.23	0.73	2.86 (4.06)	2.94 (4.40)	***0.34 (0.00)***	***1.03 (0.01)***
$\widehat{\Sigma}_A$	1503.22	306.00	104.94 (216.86)	295.19 (31.90)	***0.99 (0.00)***	***7.08 (33.01)***
$\widehat{\Sigma}_B$	1503.22	7.26	19.04 (98.85)	18.85 (5.96)	***4.78 (0.00)***	***2.16 (2.41)***
$\widehat{\Sigma}_C$	1503.22	8.90	5.19 (0.20)	6.07 (0.21)	***2.35 (0.00)***	***1.52 (0.27)***

Table 1: *Quantitative performance of the estimation for the **dataset I**.* Mean (standard deviation) relative errors (expressed as a percentage) for the estimated parameters of the GMM within the dataset I, according to the initial positions of the centroïdes and the type of algorithm. Over 100 runs.



(a) Initialisation 1 – Relative error for α (in %)

(b) Initialisation 2 – Relative error for α (in %)

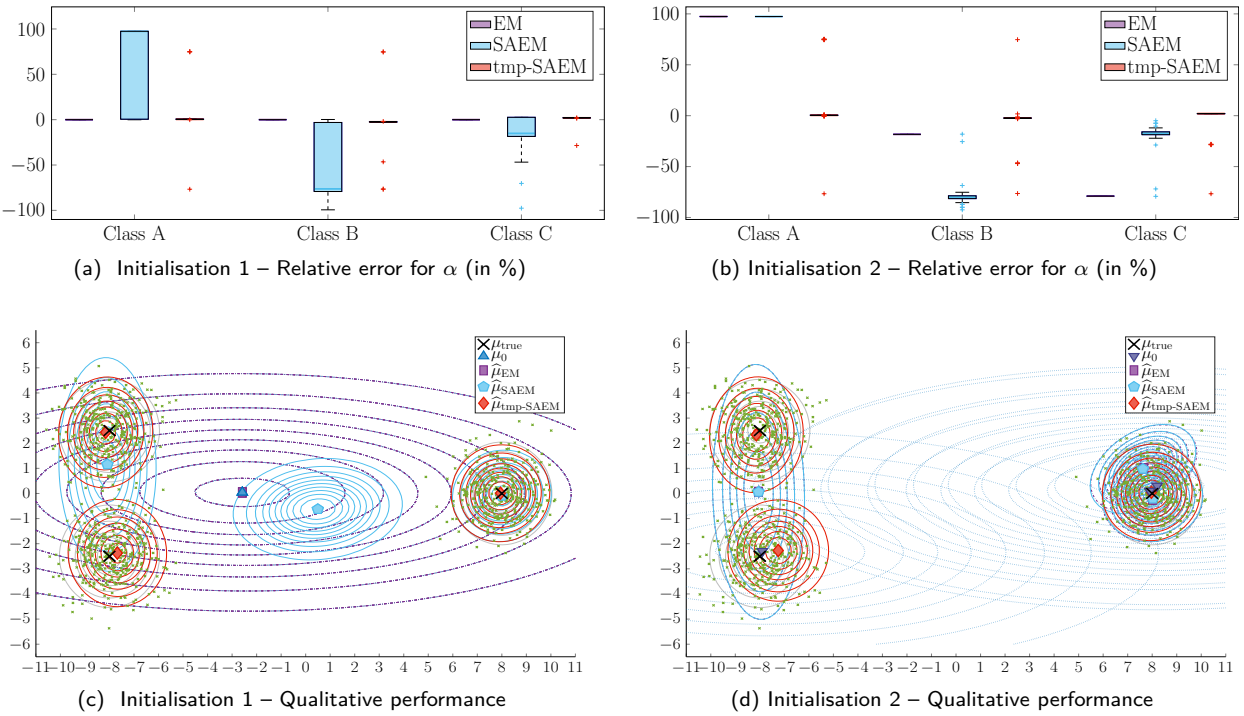
(c) Initialisation 1 – Qualitative performance

(d) Initialisation 2 – Qualitative performance

Figure 5: *Performance of the estimation for the **dataset I**.* Figs. 5a and 5b: Relative error (expressed as a percentage) for the weights α, for 100 runs and $n = 1000$, according to the type of initialization.
Figs. 5c and 5d: Average centroïdes and covariance matrices estimated by the EM, the SAEM and the tempering-SAEM algorithms within the dataset I, according to the initialization. In purple dashed lines, the covariance matrix estimated by the EM; in blue plain lines, the one estimated by the SAEM and in bold red lines the one estimated by the tempering-SAEM. In dotted blue lines the initial covariance matrices associated to the two different initial means (the blue crosses).
Tempering scheme: $a = 0$, $b = -10$, $c = 2$, $r = 10$.

	EM - 1	**EM** - 2	**SAEM** - 1	**SAEM** - 2	**tmp-SAEM** - 1	**tmp-SAEM** - 2
$\widehat{\alpha}_A$	***0.00***	97.4	55.16 (48.27)	97.4 (0.00)	0.34 (19.25)	***3.81 (18.33)***
$\widehat{\alpha}_B$	***0.00***	−18.36	−44.85 (38.38)	−79.44 (7.78)	−2.01 (18.19)	***-3.67 (12.17)***
$\widehat{\alpha}_C$	***0.00***	−79.04	−10.31 (15.68)	−17.96 (7.78)	1.67 (3.15)	***-0.14 (10.95)***
$\widehat{\mu}_A$	70.97	29.12	17.05 (13.80)	29.12 (0.00)	***3.34 (7.31)***	***3.21 (6.94)***
$\widehat{\mu}_B$	70.97	192.96	104.28 (92.43)	187.28 (1.11)	***5.31 (25.61)***	***9.47 (39.60)***
$\widehat{\mu}_C$	132.29	13.28	2.15 (1.82)	2.94 (1.09)	***0.79 (0.48)***	***1.45 (5.57)***
$\widehat{\Sigma}_A$	1438.21	154.34	88.00 (57.50)	154.34 (0.00)	***7.81 (27.00)***	***10.60 (34.47)***
$\widehat{\Sigma}_B$	1438.21	10.58	44.17 (608.01)	13.68 (0.01)	***7.28 (27.00)***	***3.48 (13.66)***
$\widehat{\Sigma}_C$	1438.21	13.60	7.64 (0.13)	10.12 (0.00)	***4.14 (0.90)***	***4.63 (2.97)***

Table 2: *Quantitative performance of the estimation for the **dataset II***. Mean (standard deviation) relative errors (expressed as a percentage) for the estimated parameters of the GMM within the dataset II, according to the initial positions of the centroïdes and the type of algorithm. Over 100 runs.



(a) Initialisation 1 – Relative error for α (in %)

(b) Initialisation 2 – Relative error for α (in %)

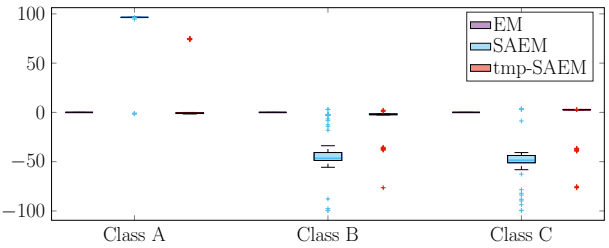
(c) Initialisation 1 – Qualitative performance

(d) Initialisation 2 – Qualitative performance

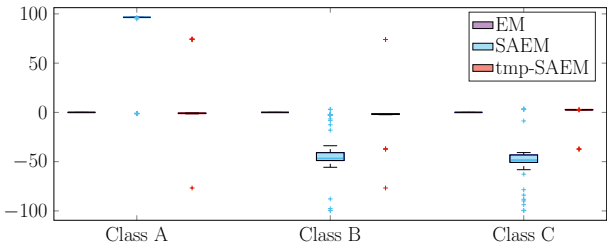
Figure 6: *Performance of the estimation for the **dataset II***. Figs. 6a and 6b: Relative error (expressed as a percentage) for the weights α, for 100 runs and $n = 1000$, according to the type of initialization.
Figs. 6c and 6d: Average centroïdes and covariance matrices estimated by the EM, the SAEM and the tempering-SAEM algorithms within the dataset II, according to the initialization. In purple dashed lines, the covariance matrix estimated by the EM; in blue plain lines, the one estimated by the SAEM and in bold red lines the one estimated by the tempering-SAEM. In dotted blue lines the initial covariance matrices associated to the two different initial means (the blue crosses).
Tempering scheme: $a = 0$, $b = -4.5$, $c = 1$, $r = 4.8$.

	EM - 1	**EM** - 2	**SAEM** - 1	**SAEM** - 2	**tmp-SAEM** - 1	**tmp-SAEM** - 2
$\widehat{\alpha}_A$	***0.00***	96.5	82.77 (34.01)	94.91 (12.40)	2.99 (22.04)	***68.43 (20.77)***
$\widehat{\alpha}_B$	***0.00***	***-24.85***	−41.62 (23.97)	−44.71 (18.13)	−4.64 (19.36)	−33.88 (9.76)
$\widehat{\alpha}_C$	***0.00***	−71.65	−41.15 (25.54)	−50.20 (18.62)	1.65 (6.42)	***-34.55 (11.24)***
$\widehat{\mu}_A$	70.56	20.82	18.11 (6.74)	20.51 (2.46)	***3.58 (7.16)***	***19.26 (5.28)***
$\widehat{\mu}_B$	70.56	196.07	158.34 (69.90)	187.25 (24.20)	***9.84 (38.15)***	***174.04 (51.80)***
$\widehat{\mu}_C$	131.10	5.64	6.46 (4.50)	7.43 (3.48)	***0.95 (1.04)***	***7.10 (1.93)***
$\widehat{\Sigma}_A$	1451.58	87.14	75.49 (8.35)	85.79 (1.10)	***10.40 (21.34)***	***80.38 (22.82)***
$\widehat{\Sigma}_B$	1451.58	***5.51***	29.18 (194.51)	12.84 (1.76)	***6.42 (11.62)***	11.61 (2.48)
$\widehat{\Sigma}_C$	1451.58	***6.99***	7.07 (0.21)	7.92 (0.21)	***3.06 (0.67)***	7.49 (1.49)

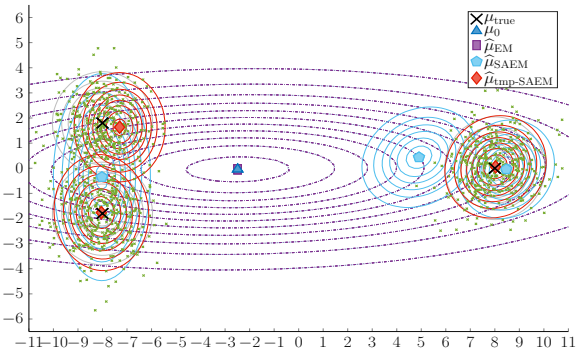
Table 3: *Quantitative performance of the estimation for the **dataset III***. Mean (standard deviation) relative errors (expressed as a percentage) for the estimated parameters of the GMM within the dataset III, according to the initial positions of the centroïdes and the type of algorithm. Over 100 runs.



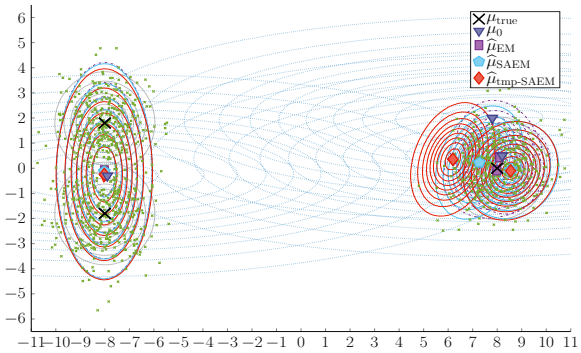
(a) Initialisation 1 – Relative error for α (in %)



(b) Initialisation 2 – Relative error for α (in %)



(c) Initialisation 1 – Qualitative performance



(d) Initialisation 2 – Qualitative performance

Figure 7: *Performance of the estimation for the **dataset III***. Figs. 7a and 7b: Relative error (expressed as a percentage) for the weights α, for 100 runs and $n = 1000$, according to the type of initialization.
Figs. 7c and 7d: Average centroïdes and covariance matrices estimated by the EM, the SAEM and the tempering-SAEM algorithms within the dataset II, according to the initialization. In purple dashed lines, the covariance matrix estimated by the EM; in blue plain lines, the one estimated by the SAEM and in bold red lines the one estimated by the tempering-SAEM. In dotted blue lines the initial covariance matrices associated to the two different initial means (the blue crosses).
Tempering scheme: $a = 0$, $b = -4.7$, $c = 1$, $r = 5$.

Still regarding initialization 2 and dataset 3, the tempering-SAEM is nevertheless at least competitive with the SAEM algorithm. Most interesting behavior: even if the tempering-SAEM does not always explain the whole distribution, the relative error may fall to zero with the tempering-SAEM, whereas this is never the case for the SAEM algorithm. In other words, the tempering-SAEM favor the convergence toward global maxima, and can almost surely reach them as in the first dataset, that is exactly the expected behavior of this algorithm.

Last, we present at Figures B.15 and B.16, in Appendix B, the evolution of the means and their associated covariance matrices. The lines 6 and 7 of Figure B.15 illustrate the capacity of the tempering-SAEM to distinguish two close classes. On the contrary, the SAEM algorithm does not seem to be able to do so and remains trapped in a local minimum.

This experiment also highlights the benefits of an oscillating temperature scheme over a simple warming up phase at the beginning of the optimization. Indeed, initializing the centroïds to the mean of the dataset may be interpreted as the limit case of heating the conditional distribution for the first iterations. However, our experiments show that the SAEM initialized at the mean (Initialization 1) behave less well than the tempering-SAEM, whatever the initialization.

4.1.3. A More Realistic Situation

Before focusing on independent factor analysis, we consider a more realistic dataset, *i.e.* a noisier and ten clusters' dataset. This third dataset is presented at Figure 8. We deliberately generated a dataset with huge overlapping (Class F is completely included in Class G for instance), given that this situation is known to not be fully explainable by GMM. Therefore, close attention will be paid to the estimation of classes with wide overlapping (Classes C *vs* D *vs* E, Classes F *vs* G and Classes G *vs* H).

The relative errors for the different parameters are displayed in Table 4; as previously, we consider the algebraic relative error for the weights α. We also provide a box plot of these relative errors in Figure 9. The EM algorithm provides a good explanation of the data set. However, it tends to strongly empty classes (Classes D and E) in favor of others (Classes A and G). Such behavior is prohibitive if we want to perform a classification task based on the estimated parameters. Moreover, even when the EM algorithm outperforms the tempering-SAEM, the latter remains competitive overall, which is generally not the case otherwise. In particular, Class I is poorly learned by the EM algorithm.

We perform the estimation 150 times for both SAEM and tempering-SAEM algorithms, with the same initial position μ_0 chosen at random within the dataset. In general, the tempering-SAEM tends to reduce the relative errors compared to the SAEM algorithm, which allows for a better explanation of the dataset. Class A, which is relatively independent of the rest of the dataset, is always well-learned. Surprisingly, the SAEM is unable to estimate Class J, which is however also independent of the rest of the dataset. Thankfully, the tempering-SAEM does not suffer from this limitation. As expected, the estimate is less accurate in the case of overlapping clusters, such as C *vs* D, F *vs* G, or G *vs* H. However, the tempering-SAEM performs better estimation of the parameters compared to the SAEM. To challenge the estimation, the data were noisy. This leads to a less-accurate estimate for the covariance matrices. Nevertheless, as before, the tempering-SAEM is still doing better than the SAEM algorithm.

4.1.4. The Weastbreast Cancer Wisconsin Dataset

Last, we want to test our algorithm on a real-life dataset, with known class labels. We chose the weastbreast cancer Wisconsin (WDBC) dataset available from UCI machine learning repository (Dua and Graff, 2017). It is commonly used in comparisons of clustering algorithms. This data set concerns breast cancer: 569 tumors are listed. For each of them, 30 nuclear features are available as well as their diagnosis: Malignant (212) or Benign (357). Some of the features in the datasets are more selective and decisive than others and certain combinations of variables are known to allow for easier classification. This is for example the case of the triplet composed of "worst" area, "worst" smoothness and mean texture. We refer to this triplet as Situation 1 in Table 5. Figure 10 displays the result of this experiment. As expected, the estimation is good for all three algorithms; the tempering SAEM is slightly more accurate.

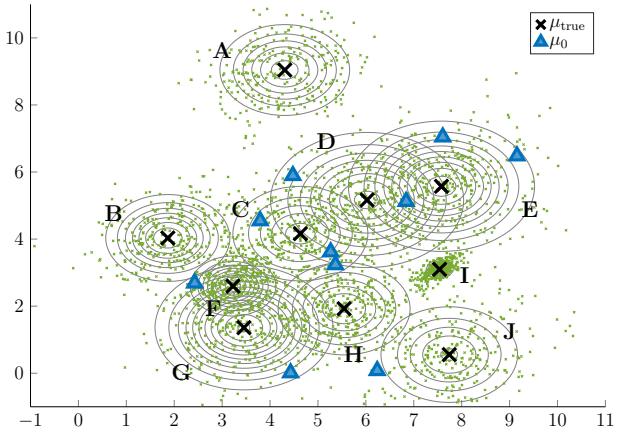
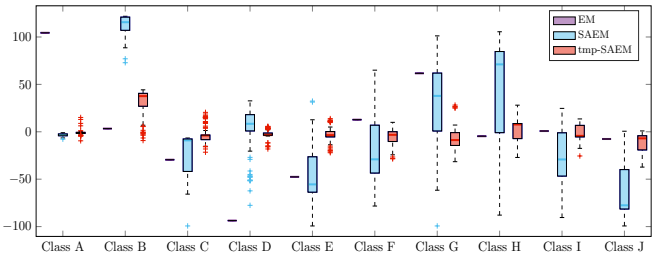


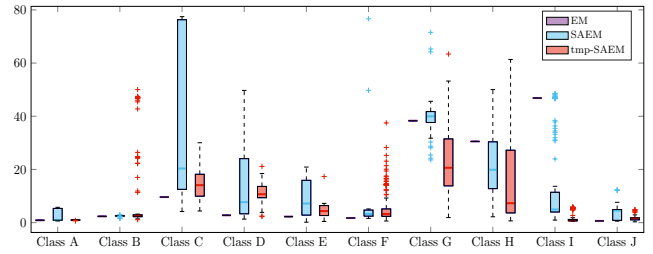
Figure 8: *Learning dataset* used to perform the experience regarding Section 4.1.3.

	EM	SAEM	tmp-SAEM
$\widehat{\alpha}_A$	104.41	−3.36 (1.09)	***-1.04 (2.48)***
$\widehat{\alpha}_B$	***3.36***	112.56 (8.46)	30.94 (14.17)
$\widehat{\alpha}_C$	−29.48	−24.99 (18.92)	***-3.12 (8.72)***
$\widehat{\alpha}_D$	−93.68	3.09 (20.10)	***-2.68 (3.61)***
$\widehat{\alpha}_E$	−47.55	−42.68 (27.14)	***-3.02 (6.58)***
$\widehat{\alpha}_F$	12.77	−18.34 (27.73)	***-5.41 (7.60)***
$\widehat{\alpha}_G$	61.66	24.91 (44.57)	***-8.08 (11.39)***
$\widehat{\alpha}_H$	***-4.68***	38.81 (54.36)	5.38 (12.96)
$\widehat{\alpha}_I$	***0.79***	−26.94 (22.99)	−1.89 (7.81)
$\widehat{\alpha}_J$	***-7.61***	−63.06 (27.95)	−11.09 (8.99)
$\widehat{\mu}_A$	***0.89***	2.98 (2.27)	0.94 (0.05)
$\widehat{\mu}_B$	***2.36***	2.52 (0.18)	9.03 (14.73)
$\widehat{\mu}_C$	***9.61***	33.36 (27.75)	14.80 (6.15)
$\widehat{\mu}_D$	***2.76***	16.28 (17.18)	11.08 (3.63)
$\widehat{\mu}_E$	***2.28***	8.18 (5.78)	4.44 (2.29)
$\widehat{\mu}_F$	***1.74***	4.21 (7.50)	6.20 (6.49)
$\widehat{\mu}_G$	38.32	39.75 (5.46)	***22.30 (11.61)***
$\widehat{\mu}_H$	30.53	20.75 (11.42)	***14.42 (14.03)***
$\widehat{\mu}_I$	46.84	13.70 (16.88)	***1.77 (1.89)***
$\widehat{\mu}_J$	***0.65***	3.52 (2.44)	1.59 (0.82)
$\widehat{\Sigma}_A$	***8.57***	7.34 (0.02)	8.71 (0.14)
$\widehat{\Sigma}_B$	***10.25***	10.70 (0.00)	17.52 (17.17)
$\widehat{\Sigma}_C$	***11.81***	15.45 (3.91)	18.30 (12.27)
$\widehat{\Sigma}_D$	17.66	***16.15 (0.22)***	21.55 (16.17)
$\widehat{\Sigma}_E$	12.82	15.81 (0.41)	***10.03 (8.53)***
$\widehat{\Sigma}_F$	***27.22***	30.81 (0.20)	32.02 (21.88)
$\widehat{\Sigma}_G$	26.81	24.21 (1.02)	***18.78 (16.30)***
$\widehat{\Sigma}_H$	***27.09***	43.85 (4.34)	31.83 (24.37)
$\widehat{\Sigma}_I$	52.11	47.44 (3.60)	***12.09 (11.56)***
$\widehat{\Sigma}_J$	***12.12***	12.80 (0.12)	12.23 (2.72)

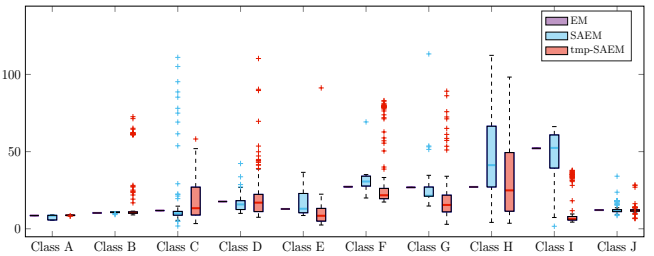
Table 4: *Quantitative performance of the estimation for the ten clusters' dataset.* Mean (standard deviation) relative errors (expressed as a percentage) for the estimated parameters of the GMM according the type of algorithm. Over 150 runs.



(a) Relative error (expressed as a percentage) for α



(b) Relative error (expressed as a percentage) for μ

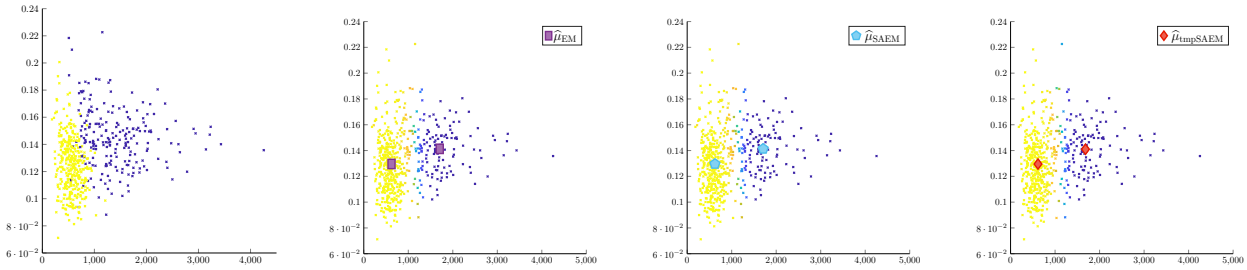


(c) KL distance between the true Σ and the estimated ones

Figure 9: Relative error (expressed as a percentage) for the weights α and the centroïds μ. Kullback-Leibler distance between the true covariance matrices Σ and the estimated ones, for 500 runs and $n = 300$.
Tempering scheme: $a = 0$, $b = -3$, $c = 2.5$, $r = 60$.

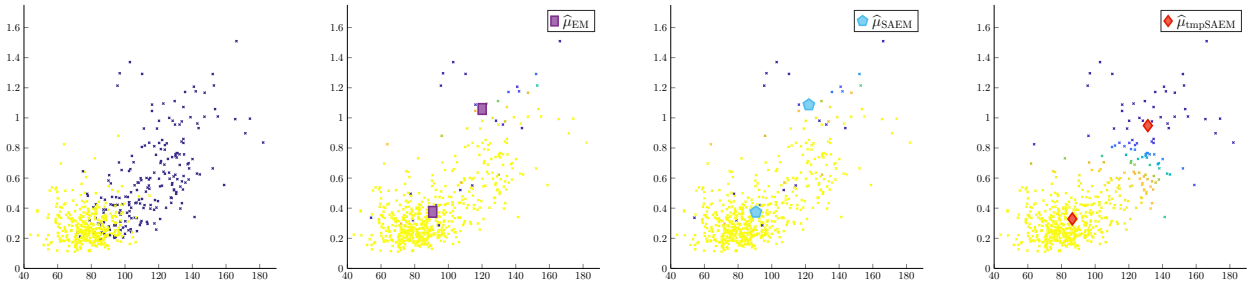
	EM	SAEM	tmp-SAEM
Situation 1	81	81 (0)	***77.93 (0.07)***
Situation 2	192	192.45 (2.09)	***149.77 (5.49)***

Table 5: Mean (standard deviation) number of mislabeled tumors, over 100 runs.



(a) Learning dataset (b) EM algorithm (c) SAEM algorithm (d) tmp-SAEM algorithm

Figure 10: *Weastbreast Cancer Wisconsin Dataset – Situation 1.* Fig. 10a: Learning dataset colored according to the true label: Malignant in yellow and Benign in dark blue. Figs. 10b, 10c and 10d: Observed data colored according to their mean probability of belonging to the Malignant class: from 0 in dark blue to 1 in yellow. Average probability taken over 100 runs. *Tempering scheme*: $a = 0$, $b = -1$, $c = 1$, $r = 1$.



(a) Learning dataset (b) EM algorithm (c) SAEM algorithm (d) tmp-SAEM algorithm

Figure 11: *Weastbreast Cancer Wisconsin Dataset – Situation 2.* Fig. 11a: Learning dataset colored according to the true label. Figs. 11b, 11c and 11d: Observed data colored according to their mean probability of belonging to the Malignant class. *Tempering scheme*: $a = 0$, $b = -1$, $c = 1$, $r = 1$.

More interestingly, we also consider another combination of variables, not known to perform as well, which brings us to Situation 2: the mean perimeter and the standard error of the radius and the symmetry. Despite a lower precision compared to situation 1, the tempering-SAEM algorithm makes it possible to counterbalance this inefficiency (Figure 11). In Table 5, the number of mislabeled tumors is obtained by imposing a 50% threshold of being of the type Malignant or Benign based on the probability of belonging to the Malignant class displayed at Figures 10 and 11.

4.2. Independent Factor Analysis

The decomposition of a sample of multi-variable data on a relevant subspace is a recurrent problem in many different fields from source separation problem in acoustic signals to computer vision and medical image analysis. Independent component analysis has become one of the standard approaches (Bell and Sejnowski, 1995). This technique relies upon a data augmentation scheme, where the (unobserved) inputs are viewed as the missing data. We observe multivariable data y which are measured by n sensors and supposed to arise from m source signals x, that are linearly mixed together by some linear transformation H, and corrupted by an additive Gaussian noise ε. Simply put, we observe $y = (y^{(t)})_{t\in[\![1,T]\!]}$, where each

measurement is a point of $\mathbb{R}^n$ and assumed to be given by $y^{(t)} = Hx^{(t)} + \varepsilon^{(t)}$, where $H \in \mathcal{M}_{n,m}\mathbb{R}$, $x^{(t)} \in \mathbb{R}^m$ and $\varepsilon^{(t)} \overset{i.i.d}{\sim} \mathcal{N}(0, \lambda I_n)$, $\lambda \in \mathbb{R}$. The suitability of the SAEM algorithm in this context has been demonstrated in (Moulines et al., 1997) and (Allassonnière and Younes, 2012). We propose here to modify the learning principle to make the procedure less susceptible to trapping states.

As in (Moulines et al., 1997) and (Attias, 1999), we assume that:

1. $(x^{(t)})_{t\in[\![1,T]\!]}$ and $(\varepsilon^{(t)})_{t\in[\![1,T]\!]}$ are independent;
2. $(x^{(t)})_{t\in[\![1,T]\!]}$ is an i.i.d sequence of random vectors, with independent component. Each component $x_i^{(t)}$ is given by a mixture of k Gaussians indexed by $z_i^{(t)} \in [\![1,k]\!]$ with means $\mu_{z_i^{(t)}}$, variances $\sigma^2_{z_i^{(t)}}$ and mixing proportions $\alpha_{z_i^{(t)}}$:

$$q(x_i^{(t)};\theta) = \sum_{z_i^{(t)}=1}^{k} \alpha_{z_i^{(t)}}\, \mathcal{G}\left(x_i^{(t)} - \mu_{z_i^{(t)}}; \sigma^2_{z_i^{(t)}}\right) \qquad \text{and} \qquad \forall j \in [\![1,k]\!], \quad \theta_j = \left(\alpha_j, \mu_j, \sigma_j^2\right),$$

where for all vectors x and μ and all symmetric matrix Σ, $\mathcal{G}(x-\mu,\Sigma)$ refers to the (multivariate) Gaussian distribution and $\theta = (\theta_j)_{j\in[\![1,k]\!]}$.

This model is called independent factor analysis (IFA). The problem is to find the value of the parameter $W = (H, \lambda, \theta)$ given y. Identifiability in this model is discussed in (Comon, 1994). Basically, the sources are defined only at a permutation of order and scaling. To avoid trivialities, we fix the variances $(\sigma_j^2)_{j\in[\![1,k]\!]}$ to one (Allassonnière and Younes, 2012). Note that this definition of the IFA model is somewhat less general than the one introduced by Attias (1999) in which the components are supposed to be independent but not necessarily identically distributed. Nevertheless, it has been shown that restrictive IFA models can perform well in practice (Allassonnière and Younes, 2012).

The likelihood of the IFA can be put in exponential form using the sufficient statistics, for all $j \in [\![1,k]\!]$,

$$S_{1,j}(x,y,z) = \frac{1}{m}\sum_{i=1}^{m} \mathbb{1}_{\{z_i=j\}}; \qquad S_{2,j}(x,y,z) = \frac{1}{m}\sum_{i=1}^{m} x_i\, \mathbb{1}_{\{z_i=j\}}; \qquad S_{3,j}(x,y,z) = \frac{1}{m}\sum_{i=1}^{m} x_i^2\, \mathbb{1}_{\{z_i=j\}};$$
$$S_4(x,y,z) = y\,{}^t y; \qquad S_5(x,y,z) = y\,{}^t x; \qquad S_6(x,y,z) = x\,{}^t x.$$

The M-step is then given by

$$H = [S_5]\left([S_6]\right)^{-1}; \qquad \alpha = [S_1]; \qquad \mu = \frac{[S_2]}{[S_1]}; \qquad \sigma^2 = \mathbf{1}_k;$$

$$\lambda = \|\,[S_6]\,\|_2^2 - 2\langle\, H \mid [S_5]\,\rangle + \langle\, {}^t\!HH \mid [S_6]\,\rangle,$$

where $\mathbf{1}_k$ stands for the k-vector off all 1 and the brackets denote the empirical-average. Moreover, it is possible to compute the conditional distribution of the hidden variable (x,z) given observed values of y and the E-step can be computed exactly Attias (1999): For all $\zeta \in [\![1,k]\!]^m$,

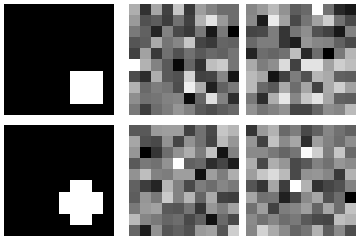
$$\mathbb{P}(z=\zeta|y;W) = \frac{\alpha_\zeta\, \mathcal{G}\left(y - H\mu_\zeta; H\Delta_\zeta\, {}^t\!H + \lambda I_n\right)}{\sum_z \alpha_z\, \mathcal{G}\left(y - H\mu_z; H\Delta_z\, {}^t\!H + \lambda I_n\right)} \qquad \text{and} \qquad q(x|y,z;W) = \mathcal{G}\left(x - \nu_{y,z}; \Sigma_z\right),$$

where

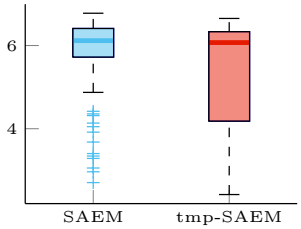
$$\alpha_z = \prod_{i=1}^{m} \alpha_{z_i}; \qquad \mu_z = (\mu_{z_i})_i; \qquad \Delta_z = \mathcal{D}iag\left((\sigma^2_{z_i})_i\right);$$

$$\Sigma_z = \left(\frac{1}{\lambda}\,{}^t\!HH + \Delta_z^{-1}\right)^{-1}; \qquad \nu_{y,z} = \Sigma_z\left(\frac{1}{\lambda}\,{}^t\!Hy + \Delta_z^{-1}\mu_z\right).$$

Thus, as well as for the GMM, we can compare the efficiency of the EM and the SAEM algorithms *versus* the tempering-SAEM, in the context of IFA. As said in the introduction, the simulated annealing

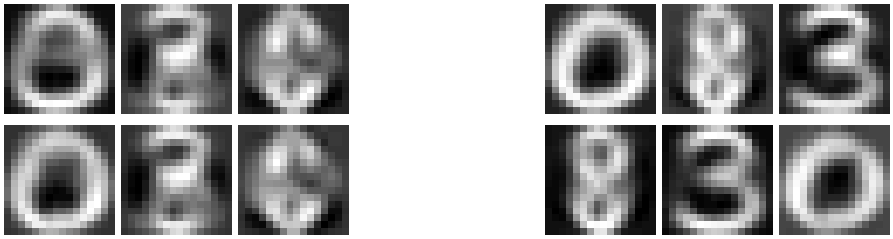


(a) Decomposition images (left) and samples of the dataset



(b) $\mathbb{L}^2$ norm

Figure 12: *Independent factor analysis – BG-ICA* renormalized $\mathbb{L}^2$ norm between the source matrix H used to build the dataset and the estimated one. The dataset consists of 100 images distributed in accordance with the two-components Bernouilli-Gaussian model build from the square and the cross binary images.



(a) SAEM (b) tempering-SAEM

Figure 13: *Independent factor analysis – USPS dataset.* Results of the independent factor estimation on a balanced mix of digits 0, 3 and 8 from the USPS database. The dataset is composed of 50 samples of each digit.

algorithm has already been successful on massive datasets in this context, such as for component separation in astrophysical images (Kuruoğlu et al., 2003). However, neither the pure simulated annealing nor the EM algorithm could fully recover the distribution.

In this section, we are going to focus rather on the SAEM algorithm than the EM one as its computational cost becomes quickly prohibitive. But, we assume that, except for the calculation time, the EM algorithm and the SAEM algorithm behave similarly. Our conviction is reinforced by the experiments of Allassonnière and Younes (2012) and ours in section 4.1.

4.2.1. Gray-Level Images – The USPS Dataset

In Section 4.1, we were interested in the performance of our algorithm for data generated according to the true model. We relax here this assumption and observe $T = 100$ images distributed under the Bernoulli-Gaussian model (BG-ICA, (Allassonnière and Younes, 2012)), with two components. The components are represented as two-dimensional binary images. The first one is a black image with a white cross in the top left corner. The second one has a white square in the bottom right corner. In Figure 12, we present the two decomposition images, 4 typical observations and the renormalized $\mathbb{L}^2$ norm between the true H (in the BG-ICA model) and the estimated one for 100 runs. Note that to the naked eye, the quality of the observed data is very similar to that of the astrophysical maps of Kuruoğlu et al. (2003): very noisy images, whose underlying structure cannot be presumed. We are therefore able to relate our experiences with theirs and to measure the contribution of a coupled SAEM/simulated annealing strategy compared to the use of only one of these two algorithms.

This experience confirms the robustness of the tempering-SAEM. Moreover, one could have feared that the augmentation of the number of hyper-parameters due to the choice of the temperature scheme would increase the variance. Figure 12 eliminates this assumption. However, the context is very favorable to the SAEM algorithm which obtain very good and hard to outperformed results.

		SAEM		tmp-SAEM	
Groups 1 *vs* 2&3	$10^{-3}\times$	0.43	(0.31)	0.37	(0.27)
Groups 1 *vs* 2	$10^{-3}\times$	11.76	(7.43)	11.33	(7.18)

Table 6: Mean (standard deviation) of the p-values for the five decomposition vectors presented at Figure 14, over 50 runs.

To measure the efficiency of the tempering-SAEM, we test it on the USPS database, which contains gray-level images of handwritten digits. We consider a balanced mixture of the digits 0, 3 and 8, which consists of 50 samples for each of the three digits. We then run both the SAEM and the tempering-SAEM. We present at Figure 13 two typical runs (inline). If both succeed in discriminating 0 against 3 and 8, the tempering-SAEM outperforms the SAEM algorithm concerning 3 versus 8. Thus, the tempering-SAEM produces meaningful sources, which could be the result of a clustering procedure, while the SAEM runs into difficulties. Hence, this experience suggests that the tempering-SAEM can indeed escape from local *maxima* in which the SAEM can be trapped.

4.2.2. Hippocampi Surfaces

Last, we consider a dataset consisting of 101 hippocampi surfaces. The subjects of the dataset can be split into three groups of size 57, 32 and 12 respectively. The first group corresponds to healthy patients; the next two groups correspond to patients with Alzheimer's disease, at two stages of advancement (mild and advanced). Over each hippocampus, a scalar field represents the deformation of the considered hippocampus regarding a template one. Thus we can study the diversity of atrophy patterns, depending on the patient's state of health. We have computed $m = 5$ decomposition vectors based on the complete data set. Figure 14 presents these decomposition vectors mapped on the meshed hippocampus for both SAEM and tempering-SAEM algorithms. For comparison purposes, we enforce the same colorbar for both experiments and all hippocampi. Then, it seems that the two algorithms behave in much the same way, at least visually. This experiment attests to the reliability of the tempering-SAEM.

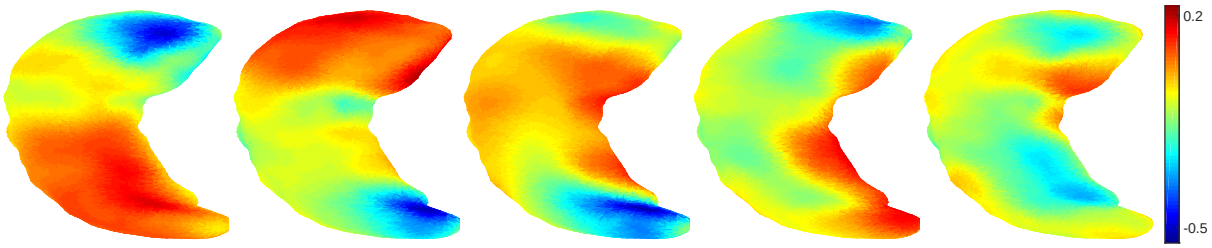
In Table 6, we provide the p-values obtained from the comparison of the five columns of H among the three subgroups. The test is based on a Hoteling T-statistic evaluated on the coefficients, the p-value is computed using permutation sampling. Following Allassonnière and Younes (2012), we compute the p-value for two different comparisons: the first one compares the healthy patients with respect to Alzheimer's and dementia patients (the two last groups). The second test compares the healthy patient with respect to the mild Alzheimer's patients (the second group). Due to the stochasticity of the SAEM algorithm, we computed an average and a standard deviation of the p-values over 50 runs, with the same initial conditions. Thus, the tempering-SAEM algorithm always behaves at least as well as the SAEM algorithm.

Finally, applying the tempering-SAEM for independent factor analysis aims to check that the advantages of the tempering-SAEM over the SAEM can improve or at least does not deteriorate the results of maximum likelihood estimation in complex hierarchical models.

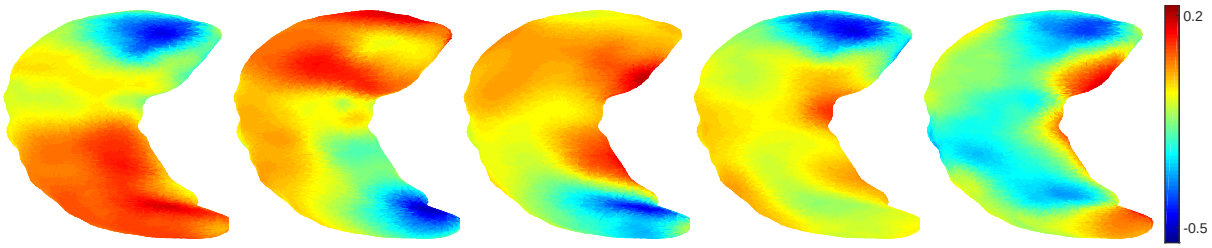
4.3. Discussion and Perspective

We propose here a new stochastic approximation version of the EM algorithm. The benefit of this general procedure is twofold: we can deal with the problem of intractable or difficult sampling in one hand and favor convergence toward global *maxima* on the other hand.

Our first contribution is theoretical with the proof of the convergence of the approximated-SAEM toward stationary points. This proof, due to its similarity with the corresponding one for the SAEM, highlights the unconstraining nature of our algorithm and its great applicability. In particular, under the same assumptions about the model's likelihood as those introduced by Delyon et al. (1999), the approximated-SAEM converges towards local *maxima*. This result gives an *a posteriori* justification for some existent schemes like the ABC-SAEM (Picchini and Samson, 2018) or MONOLIX (Lavielle, 2014). Moreover, our general framework is versatile enough to encompass a wide range of algorithms. Our second contribution goes this way by proposing an instantiation of this general procedure to prevent convergence toward local *maxima*, referred to as tempering-SAEM. This tempering-SAEM method is the one used in the MONOLIX software. We have



(a) SAEM



(b) tempering-SAEM

Figure 14: *Independent factor analysis – Hippocampi dataset.* Results of the independent factor estimation on a corpus of 101 hippocampi. Atrophy patterns of the hippocampi in the context of Alzheimer's disease.

applied this algorithm in both synthetic and real data frameworks and obtained improved results with respect to the state of the art algorithms in both cases.

This opens up new perspectives. Typically, now that we have ensured the convergence of the approximated-SAEM, a natural opening concerns the study of the convergence of the approximated-MCMC-SAEM. Indeed, although the convergence of this algorithm has not yet been demonstrated, the tempering-MCMC-SAEM has already shown its numerical efficiency, especially in the case of medical applications (Debavelaere et al., 2019).

Appendix A. Theorem 2 and Lemma 2 of Delyon et al. (1999)

In order our article to be more self-contained, we recall Theorem 2 and Lemma 2 of Delyon et al. (1999). Actually, the proof of Theorem 3 is based on this theorem which establish the convergence of Robin-Monroe type approximation procedure, *i.e.* the convergence of sequences defined recursively as

$$\forall k \in \mathbb{N}, \qquad s_k \;=\; s_{k-1} + \gamma_k\big(h(s_k) + r_k + e_k\big)\,.$$

Theorem 4 (Delyon et al. (1999))**.** *Assume that*

(SA0) *With probability 1, for all* $k \in \mathbb{N}$, $s_k \in \mathcal{S}$.

(SA1) $(\gamma_k)_{k\in\mathbb{N}^*}$ *is a decreasing sequence of positive numbers such that* $\sum_{k=1}^{\infty}\gamma_k = \infty$.

(SA2) *The vector field* h *si continuous on* $\mathcal{S}$ *and there exists* $V : \mathcal{S} \to \mathbb{R}$ *continuously differentiable such that :*

(i) for all $s \in \mathcal{S}$, $F(s) = \langle \mathrm{d}_s V(z) \,|\, h(s) \rangle \leqslant 0$,
(ii) $int\,(V(\mathcal{L})) = \varnothing$ *where* $\mathcal{L} = \{s \in \mathcal{S} | F(s = 0)\}$.

(SA3) *With probability 1,* $clos\,(\{s_k\}_{k\in\mathbb{N}})$ *is a compact subset of* $\mathcal{S}$.

(SA4) *With probability 1,* $\sum \gamma_k e_k$ *exists and is finite,* $\lim r_k = 0$.

Then, with probability 1, $\overline{\lim}\, d(s_k, \mathcal{L}) = 0$.

Lemma 2 (Delyon et al. (1999))**.** *Assume (M1 - 5) and (SAEM2). Then (SA2) is satisfied with* $V = -\ell \circ \hat{\theta}$. *Moreover,*

$$\{s \in \mathcal{S} | F(s) = 0\} = \{s \in \mathcal{S} |\, \mathrm{d}_s V(s) = 0\} \qquad \text{and} \qquad \hat{\theta}\left(\{s \in \mathcal{S} | F(s) = 0\}\right) = \{\theta \in \Theta |\, \mathrm{d}_\theta \ell(\theta) = 0\}\,,$$

where $F \colon s \mapsto \langle\, \mathrm{d}_s V(s) \,|\, h(s) \,\rangle$.

Appendix B. Multivariate Gaussian Mixture Model

We give here some details about the estimation procedure in the multivariate Gaussian mixture model. The complete log-likelihood of the GMM model is

$$\log q(y, z; \theta) = -n \log 2\pi - \sum_{j=1}^{m} \sum_{i=1}^{n} \left(\frac{1}{2} \log|\Sigma_j| - \log \alpha_j + {}^t(y_i - \mu_j)\, \Sigma_j^{-1}\, (y_i - \mu_j) \right) \mathbb{1}_{\{z_i = j\}}\,.$$

Appendix B.1. Estimation through the EM Algorithm

Let k index the current iteration. The general EM algorithm iterates the following two steps:

E-step: Compute $Q(\theta|\theta^k) = \mathbb{E}\left[\log q(y, z; \theta) | y, \theta^k\right]$;

M-step: Set $\theta^{k+1} \in \operatorname{argmax}_{\theta \in \Theta} Q(\theta|\theta^k)$.

For all $(i, j) \in [\![1, n]\!] \times [\![1, m]\!]$, set $\tau_{i,j} = \mathbb{P}\left[z_i = j | y_i, \theta^k\right]$. Then,

$$Q(\theta|\theta^k) = -n \log 2\pi - \sum_{j=1}^{m} \sum_{i=1}^{n} \left(\frac{1}{2} \log|\Sigma_j| - \log \alpha_j + {}^t(y_i - \mu_j)\, \Sigma_j^{-1}\, (y_i - \mu_j) \right) \tau_{i,j}^k\,.$$

According to Bayes' rule,

$$\tau_{i,j}^k = \frac{\alpha_j^k\, \mathcal{G}(y_i - \mu_j^k; \Sigma_j^k)}{\sum_{j=1}^{m} \alpha_j^k\, \mathcal{G}(y_i - \mu_j^k; \Sigma_j^k)}\,,$$

where $\mathcal{G}(y - \mu; \Sigma)$ refers to the Gaussian distribution with mean μ and covariance matrix Σ. Lastly, a straightforward computation gives

$$\alpha_j^{k+1} = \frac{1}{n} \sum_{i=1}^{n} \tau_{i,j}^k\,, \qquad \mu_j^{k+1} = \frac{\sum_{i=1}^{n} \tau_{i,j}^k y_i}{\sum_{i=1}^{n} \tau_{i,j}^k} \qquad \text{and} \qquad \Sigma_j^{k+1} = \frac{\sum_{i=1}^{n} \tau_{i,j}^k (y_i - \mu_j^{k+1})\, {}^t(y_i - \mu_j^{k+1})}{\sum_{i=1}^{n} \tau_{i,j}^k}\,.$$

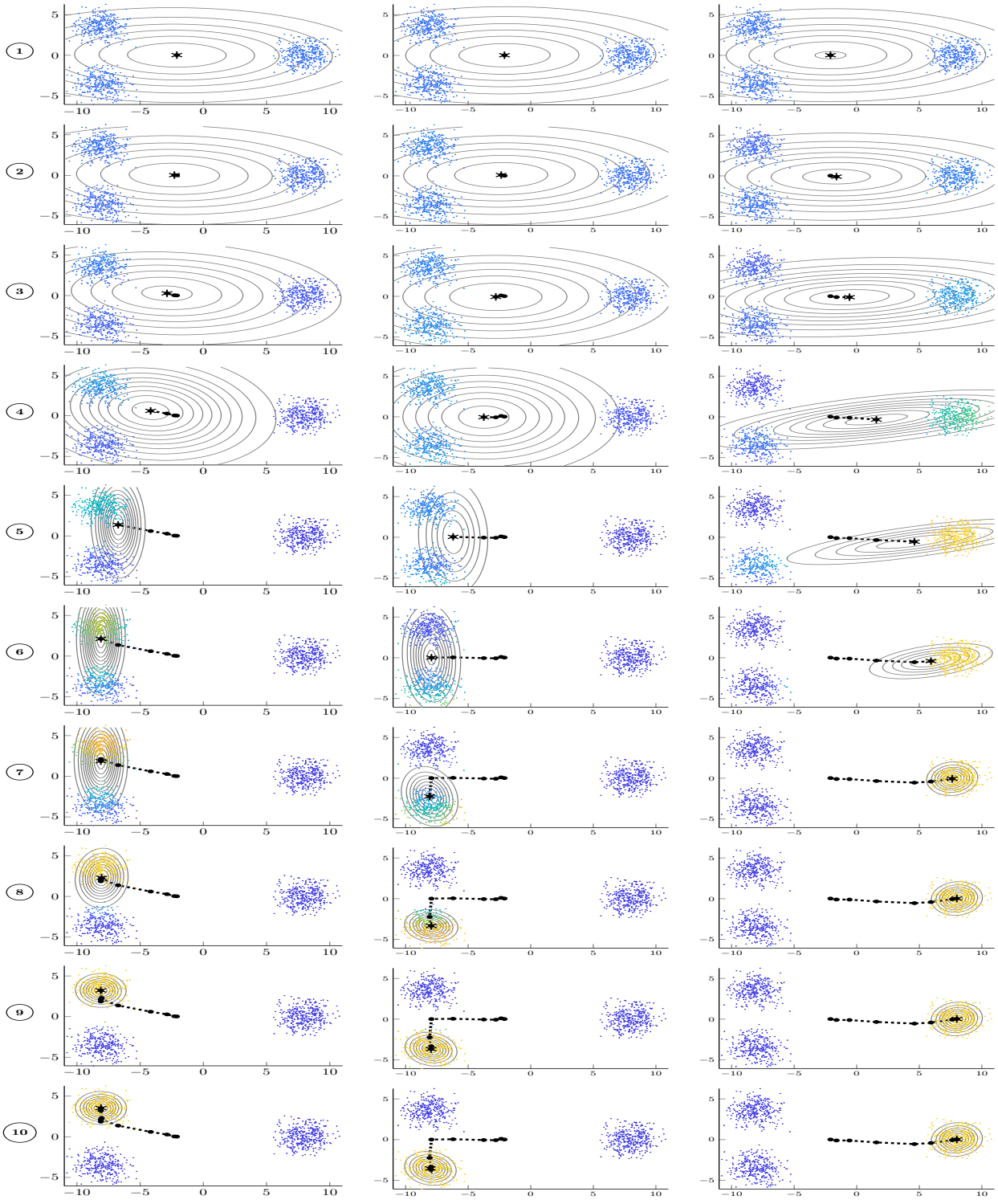
Appendix B.2. Estimation through the SAEM Algorithm

Given a sequence of positive step-size for the stochastic approximation $\gamma = (\gamma^k)_{k \in \mathbb{N}}$, the general SAEM algorithm iterates the following two steps:

SAE-step: Sample *a* new hidden variable z^{k+1} according to the conditional distribution $q(\cdot|y, \theta^k)$ and compute

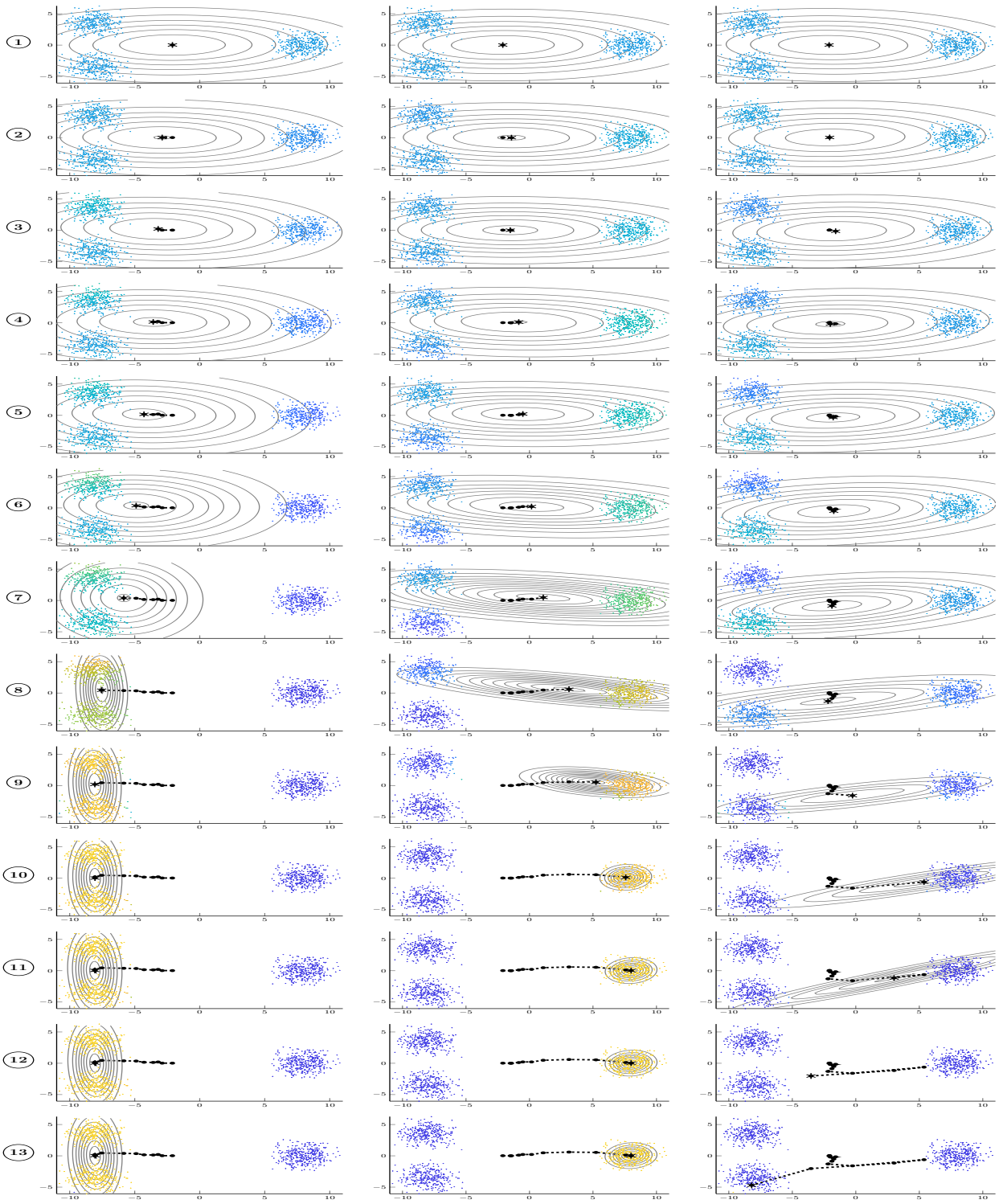
$$Q^{k+1}(\theta) = Q^k(\theta) + \gamma^k \left(\log q(y, z^{k+1}; \theta^k) - Q^k(\theta) \right)\,;$$

M-step: Set $\theta^{k+1} \in \operatorname{argmax}_{\theta \in \Theta} Q^{k+1}(\theta)$.



(a) Classe A (b) Classe B (c) Classe C

Figure B.15: *Evolution of the parameters throughout the estimation by the tempering-SAEM algorithm*, within the dataset I, with initial means at the barycenter of the dataset, for a typical run. For each class, we plot the trajectories of the mean μ and the evolution of the associated covariance. The observed data are colored according to their probability of belonging to the classes: from 0 in dark blue to 1 in yellow.



(a) Classes A & B (b) Classe C (c) Empty classe

Figure B.16: *Evolution of the parameters throughout the estimation by the SAEM algorithm*, within the dataset I, with initial means at the barycenter of the dataset, for a typical run. We keep the conventions of the previous figure.

The GMM belongs to the curved exponential family. Actually, for all y, z and θ,

$$\log q(y,z;\theta) \;=\; -n\log(2\pi) + \sum_{j=1}^{m}\left(\log\alpha_j - \frac{1}{2}\log|\Sigma_j| + \langle\, \mu_j\, {}^t\mu_j \,|\, \Sigma_j^{-1}\,\rangle_{\mathcal{F}}\right) S_{1,j}(y,z) \\ + \sum_{j=1}^{m}\left[\,\langle\, \Sigma_j^{-1} \,|\, S_{3,j}(y,z)\,\rangle_{\mathcal{F}} - 2\langle\, \Sigma_j^{-1}\mu_j \,|\, S_{2,j}(y,z)\,\rangle\,\right],$$

where, for all $j \in [\![1,m]\!]$,

$$S_{1,j}(y,z) \;=\; \sum_{ix=1}^{n} \mathbb{1}_{z_i=j}\;; \qquad S_{2,j}(y,z) \;=\; \sum_{i=1}^{n} y_i\,\mathbb{1}_{z_i=j}\;; \qquad S_{3,j}(y,z) \;=\; \sum_{i=1}^{n} y_i\,{}^t y_i\,\mathbb{1}_{z_i=j}\,.$$

So, the SAE-step is replaced by an update of the estimation of the conditional expectation of the sufficient statistics, namely, for all $\ell \in \{1,2,3\}$, and all j,

$$S_{\ell,j}^{k+1} \;=\; S_{\ell,j}^{k} + \gamma^k\left(S_{\ell,j}(y,z^{k+1}) - S_{\ell,j}^{k}\right)\,,$$

where, for all i, z_i^{k+1} is sampled from the discrete law $\sum_{j=1}^{m}\tau_{i,j}^k\delta_j$, where $\tau_{i,j}^k = \mathbb{P}\left[z_i = j | y_i, \theta^k\right]$ as in the EM-case.

The M-step can also be computed in close-form:

$$\alpha_j^{k+1} \;=\; \frac{1}{n}S_{1,j}\;; \qquad \mu_j^{k+1} \;=\; \frac{S_{2,j}}{S_{1,j}}\;; \qquad \Sigma_j^{k+1} \;=\; \frac{S_{3,j} - S_{2,j}\,{}^t(\mu_j^{k+1})}{S_{1,j}}\,.$$

Appendix B.3. Estimation through the tempering-SAEM Algorithm

The previous computation remain true except that the hidden variables z_i^{k+1} are now sampled from the tempered conditional distribution $\dfrac{1}{c(T^k)}\displaystyle\sum_{j=1}^{m}{\tau_{i,j}^k}^{1/T^k}\delta_j$, where $c(T^k) = \displaystyle\sum_{j=1}^{m}{\tau_{i,j}^k}^{1/T^k}$ and T^k is defined in Section 3.2.

To stabilize the convergence of both SAEM and tempering-SAEM, we may use inverse Wishart priors for the variances and Gaussian priors for the weights.

References

Allassonnière, S., Kuhn, E., Trouvé, A., 2010. Construction of bayesian deformable models via a stochastic approximation algorithm: a convergence study. Bernoulli 16, 641–678.

Allassonnière, S., Younes, L., 2012. A stochastic algorithm for probabilistic independent component analysis. The Annals of Applied Statistics 6, 125–160.

Andrieu, C., Moulines, É., Priouret, P., 2006. Stability of stochastic approximation under verifiable conditions. SIAM Journal on Control and Optimization 44, 283–312.

Attias, H., 1999. Independent factor analysis. Neural Computation 11, 803–851.

Balakrishnan, S., Wainwright, M.J., Yu, B., 2017. Statistical guarantees for the em algorithm: From population to sample-based analysis. The Annals of Statistics 45, 77–120.

Bell, A.J., Sejnowski, T.J., 1995. An information-maximization approach to blind separation and blind deconvolution. Neural computation 7, 1129–1159.

Biernacki, C., Celeux, G., Govaert, G., 2003. Choosing starting values for the EM algorithm for getting the highest likelihood in multivariate Gaussian mixture models. Computational Statistics & Data Analysis 41, 561 – 575.

Blei, D.M., Kucukelbir, A., McAuliffe, J.D., 2017. Variational inference: A review for statisticians. Journal of the American Statistical Association 112, 859–877.

Booth, J.G., Hobert, J.P., 1999. Maximizing generalized linear mixed model likelihoods with an automated Monte Carlo EM algorithm. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 61, 265–285.

Cappé, O., Moulines, É., Rydén, T., 2005. Inference in Hidden Markov Models. Springer Series in Statistics, Springer.

Celeux, G., Diebolt, J., 1985. The SEM algorithm: a probabilistic teacher algorithm derived from the EM algorithm for the mixture problem. Computational statistics quarterly 2, 73–82.

Chan, P.L., Jacqmin, P., Lavielle, M., McFadyen, L., Weatherley, B., 2011. The use of the SAEM algorithm in MONOLIX software for estimation of population pharmacokinetic-pharmacodynamic-viral dynamics parameters of maraviroc in asymptomatic HIV subjects. Journal of pharmacokinetics and pharmacodynamics 38, 41–61.

Comon, P., 1994. Independent component analysis, a new concept? Signal Processing 36, 287 – 314. Higher Order Statistics.

Debavelaere, V., Bône, A., Durrleman, S., Allassonnière, S., 2019. Clustering of longitudinal shape data sets using mixture of separate or branching trajectories. To appear in MICAI 2019.

Delyon, B., Lavielle, M., Moulines, E., 1999. Convergence of a stochastic approximation version of the EM algorithm. The Annals of Statistics 27, 94–128.

Dempster, A., Laird, N.M., Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society. Series B 39, 1–38.

Dua, D., Graff, C., 2017. UCI machine learning repository. URL: http://archive.ics.uci.edu/ml.

Efron, B., 1978. The geometry of exponential families. The Annals of Statistics 6, 362–376.

Fort, G., Moulines, E., 2003. Convergence of the monte carlo expectation maximization for curved exponential families. The Annals of Statistics 31, 1220–1259.

Hall, P., Heyde, C.C., 1980. Martingale limit theory and its application. Probability and mathematical statistics, Academic Press.

Jank, W., 2005. Quasi-Monte Carlo sampling to improve the efficiency of Monte Carlo EM. Computational statistics & data analysis 48, 685–701.

Jank, W., 2006. Implementing and diagnosing the stochastic approximation EM algorithm. Journal of Computational and Graphical Statistics 15, 803–829.

Jiang, W., Josse, J., Lavielle, M., Gauss, T., 2018. Stochastic approximation em for logistic regression with missing values. arXiv preprint arXiv:1805.04602 .

Jordan, M.I., Ghahramani, Z., Jaakkola, T.S., Saul, L.K., 1999. An introduction to variational methods for graphical models. Machine learning 37, 183–233.

Karimi, B., Lavielle, M., Moulines, E., 2020. f-saem: A fast stochastic approximation of the em algorithm for nonlinear mixed effects models. Computational Statistics & Data Analysis 141, 123–138.

Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P., 1983. Optimization by simulated annealing. science 220, 671–680.

Kuhn, E., Lavielle, M., 2004. Coupling a stochastic approximation version of em with an mcmc procedure. ESAIM: Probability and Statistics 8, 115–131.

Kuhn, E., Lavielle, M., 2005. Maximum likelihood estimation in nonlinear mixed effects models. Computational statistics & data analysis 49, 1020–1038.

Kuruoğlu, E.E., Bedini, L., Paratore, M.T., Salerno, E., Tonazzini, A., 2003. Source separation in astrophysical maps using independent factor analysis. Neural Networks 16, 479–491.

Lavielle, M., 2014. Mixed effects models for the population approach: models, tasks, methods and tools. CRC press.

Lavielle, M., Mentré, F., 2007. Estimation of population pharmacokinetic parameters of saquinavir in HIV patients with the MONOLIX software. Journal of pharmacokinetics and pharmacodynamics 34, 229–249.

Lavielle, M., Moulines, E., 1997. A simulated annealing version of the EM algorithm for non-Gaussian deconvolution. Statistics and Computing 7, 229–236.

Marin, J.M., Pudlo, P., Robert, C.P., Ryder, R.J., 2012. Approximate Bayesian computational methods. Statistics and Computing 22, 1167–1180.

McLachlan, G., Peel, D., 2000. Finite Mixture Models. Wiley Series in Probability and Statistics, Wile.

McLachlan, G.J., Krishnan, T., 2007. The EM algorithm and extensions. volume 382. John Wiley & Sons.

Melnykov, V., Maitra, R., 2010. Finite mixture models and model-based clustering. Statistics Surveys 4, 80–116.

Meng, X.L., Van Dyk, D., 1997. The em algorithm—an old folk-song sung to a fast new tune. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 59, 511–56.

Moulines, E., Cardoso, J.F., Gassiat, E., 1997. Maximum likelihood for blind separation and deconvolution of noisy signals using mixture models, in: Acoustics, Speech, and Signal Processing, IEEE. pp. 3617–3620.

Picchini, U., Samson, A., 2018. Coupling stochastic EM and approximate Bayesian computation for parameter inference in state-space models. Computational Statistics 33, 179–212.

Robbins, H., Monro, S., 1951. A stochastic approximation method. The annals of mathematical statistics , 400–407.

Samson, A., Lavielle, M., Mentré, F., 2006. Extension of the SAEM algorithm to left-censored data in nonlinear mixed-effects model: Application to HIV dynamics model. Computational Statistics & Data Analysis 51, 1562–1574.

Titterington, D.M., Smith, A.F., Makov, U.E., 1985. Statistical analysis of finite mixture distributions. Wiley.

Wainwright, M.J., Jordan, M.I., 2008. Graphical models, exponential families, and variational inference. Foundations and Trends® in Machine Learning 1, 1–305.

Wei, G.C., Tanner, M.A., 1990. A Monte Carlo implementation of the EM algorithm and the poor man's data augmentation algorithms. Journal of the American statistical Association 85, 699–704.

Wu, C.F.J., 1983. On the convergence properties of the EM algorithm. The Annals of Statistics 11, 95–103.

Xu, L., Jordan, M.I., 1996. On convergence properties of the em algorithm for gaussian mixtures. Neural computation 8, 129–151.